LICENSE PLATE RECOGNITION SYSTEM

BY

MOHAMMED JAMEEL AHMED

A Thesis Presented to the

DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER SCIENCE

June 2003

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SUADI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis written by **MOHAMMED JAMEEL AHMED** under the direction of his thesis advisor and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER SCIENCE.**

Thesis Committee

Dr. Muhammad Sarfraz (Advisor)

Dr. Abdelmalek Zidouri (Co-Advisor)

Kanaan Faisal

Dr. Kanaan A.Faisal,
Department Chairman

M.R.K. Krishna Rao

Dr. M.R.K Krishna Rao(Member)

Dr. Wasfi Ghassan Al-Khatib (Member)

Prof. Osama A.Jannadi
Dean of Graduate Studies

Dr. Samer Nassif (Member)

17/9/2003
Date

DEANSHIP OF GRADUATE STUDIES

Dedication

This thesis is dedicated to my beloved

parents, brother, sister,

and

all my teachers.

Acknowledgment

In the name of Allah, Most Gracious, Most Merciful

All praise and glory to Almighty Allah (Subhanahu Wa Taalaa) who gave me courage and patience to carry out this work. Peace and blessing of Allah be upon last Prophet Muhammad (Peace Be Upon Him)

Acknowledgment is due to King Fahd University of Petroleum & Minerals for supporting this research.

My unrestrained appreciation goes to my advisor, Dr. Muhammad Sarfraz, for all the help and support he has given me throughout the course of this work and on several other occasions. I simply cannot begin to imagine how things would have proceeded without his help, his support, and his patience. I wish to thank my thesis co-advisor, Dr. Abdelmalek Zidouri for his suggestions. I also wish to thank my thesis committee members, Dr. M.R.K. Krishna Rao, Dr. Wasfi Ghassan Al-Khatib and Dr. Samer Nassif for their help, support, and contributions. Lastly, I thank Dr. Kalim Qureshi for his support and encouragement during the last two semesters.

I also acknowledge my colleagues and friends as I had a pleasant, enjoyable and fruitful company with them. Specially, I would like to thank my friends Akhtar Ghazi, Nazim, Farooq, Imran, Mujtaba, Arif Vantala, Moin for their continuous encouragement and support.

Special Thanks are due to senior faculty members of Indian community, especially Mohsin Siddiqui, Mir Ahmed Ali Shajee, Iqbal Basha for their timely help and providing a friendly environment.

Thanks are due to my flat mates, Khusro Mohiuddin, Iftekhar, Farrukh, Mohibuddin, Munawer, Humayun, Samiullah, Atif and Haseeb for providing me a wonderful company.

Finally, I wish to express my gratitude to my family members for being patient with me and offering words of encouragements to spur my spirit at moments of depression.

Table of Contents

List of Figures

List of Tables

Thesis Abstract

NAME: MOHAMMED JAMEEL AHMED

TITLE: LICENSE PLATE RECOGNITION SYSTEM

DEGREE MASTER OF SCIENCES

MAJOR FIELD: COMPUTER SCIENCE

DATE OF DEGREE: JUNE 2003

A License Plate Recognition (LPR) System is one kind of an Intelligent Transport System and is of considerable interest because of its potential applications in highway electronic toll collection and traffic monitoring systems. This type of applications puts high demands on the reliability of an LPR System. A lot of work has been done regarding LPR systems for Korean, Chinese, European and US license plates that generated many commercial products. However, little work has been done for Arabian LPR systems.

This thesis addresses the problem of an LPR system for Saudi Arabian plates. A general LPR system consists of four stages, Image acquisition, License plate extraction, License plate segmentation and License Plate Recognition. In this thesis we have proposed algorithms for the extraction stage based on vertical edge matching. The segmentation stage is performed using the horizontal and vertical projection profile. Finally, we have investigated two approaches to carry out the recognition stage, viz. syntactic and neural network.

The Proposed system has been implemented using MATLAB6.1 and results have shown that the syntactic approach has given better results than the neural network approach. The performance of the system has been investigated on real images of about 310 vehicles captured under various illumination conditions. Recognition of about 96% using syntactic approach shows that the system is quite efficient.

Thesis Abstract

خلاصة الرسالة

الاسم: محمد جميل أحمد

العنوان: نظام حاسوبي لقراءة لوحات ترخيص السيارات

الدرجة: الماجستير في العلوم

التخصص الرئيسي: علوم الحاسب الآلي

تاريخ التخرج: يونيو 2003

تعد نظم القراءة الحاسوبية للوحات ترخيص السيارات(License Plate Recognition Systems) من إحدى نظم المواصلات الذكية والتي تكتسب أهمية خاصة في تطبيقات جمع تعرفة الطرق السريعة و أنظمة مراقبة المرور، ويتطلب هذا النوع من الأنظمة الحاسوبية درجة عالية من الدقة ، لذا يوجد الكثير من الأبحاث العلمية التي تناولت هذه النظم بالدراسة وتم تطبيقها على لوحات ترخيص كورية وصينية وأوروبية وأمريكية ، وقد تم تصميم وإنتاج هذه النظم على نطاق تجاري بناء على هذه الأبحاث ، أما بالنسبة للوحات الترخيص العربية فلا يوجد من هذه الدراسات إلا النزر اليسير.لذلك ومن خلال هذه الرسالة قمنا بدراسة تهدف إلى تصميم نظام لقراءة لوحات ترخيص السيارات في المملكة العربية السعودية ،هذا و يتكون أي نظام حاسوبي لقراءة لوحات ترخيص السيارات من أربعة مراحل أساسية.

إحراز الصورة (image acquisition) واستخلاص اللوحة (license plate extraction) وتجزئة اللوحة (license plate segmentation) ثم مرحلة قراءة اللوحة (license plate recognition) ، وقد قمنا من خلال هذه الرسالة باقتراح طرق حاسوبية (algorithms) لاستخلاص اللوحة اعتمادا على مطابقة الحواف الرأسية (vertical edge matching) ، كما قمنا باستخدام صور الإسقاط الأفقية والرأسية (horizontal and vertical projection profiles) لإتمام مرحلة تجزئة اللوحة ، أخيرا قمنا بدراسة منهجين مختلفين هما الطريقة التركيبية (syntactic approach) وطريقة استخدام الشبكات العصبية (neural network approach) للقيام بعملية التعرف النهائية وقراءة لوحة الترخيص.

هذا وقد تم بناء تطبيق حاسوبي باستخدام برنامج ماتلاب نسخة 6.1 (matlab 6.1) كنظام لقراءة لوحات ترخيص السيارات السعودية حيث أظهرت النتائج تفوق الطريقة التركيبية على طريقة الشبكات العصبية في دقة قراءة اللوحات ، علما بأنه قد تم اختبار هذا النظام على صور 310 سيارة تم التقاطها في ظروف أضاءة مختلفة ، وقد أظهرت النتائج القدرة على قراءة لوحات الترخيص بصورة صحيحة في 96% من هذه الصور باستخدام الطريقة التركيبية.

CHAPTER 1

INTRODUCTION

License plate recognition (LPR) is an image-processing technology used to identify vehicles by their license plates. This technology is gaining popularity in security and traffic installations. Much research has already been done for the recognition of Korean, Chinese, European and other license plates, however this work seems to be the first attempt towards the recognition of Saudi Arabian license plates. This thesis presents a license plate recognition system as an application of computer vision. Computer vision is a process of using a computer to extract high level information from a digital image. This chapter will set the scene by first presenting some applications of a license plate recognition system. Next, we discuss the elements that are commonly used in a license plate recognition system. Following this, the working of a typical LPR system is described. Next, we present the structure of proposed license plate recognition system. Finally, the objectives of the work are stated. The chapter ends with a brief overview of the rest of this thesis.

1.1 Applications of LPR Systems

Vehicle license plate recognition is one form of automatic vehicle identification system. LPR systems are of considerable interest, because of their potential applications to areas such as highway electronic toll collection, automatic parking attendant, petrol station forecourt surveillance, speed limit enforcement, security, customer identification enabling personalized services, etc. Real time LPR plays a major role in automatic monitoring of traffic rules and maintaining law enforcement on public roads. This area is challenging because it requires an integration of many computer vision problem solvers, which include Object Detection and Character Recognition [23]. The automatic identification of vehicles by the contents of their license plates is important in private transport applications. There are many applications of such recognition systems, some of them are discussed below:

- **Law Enforcement:** The plate number is used to produce a violation fine on speeding vehicles, illegal use of bus lanes, and detection of stolen or wanted vehicles. Figure 1.1 is an example of a speeding car caught by the traffic camera. The rear part of the vehicle is extracted off the filmed image and is given to the system for processing. The processed result is fed into the database as input. The violators can pay the fine online and can be presented with the image of the car as a proof along with the speeding information.

Figure 1.1 Car violating the speed

- **Parking:** The LPR system is used to automatically enter pre-paid members and calculate parking fee for non-members (by comparing the exit and entry times). Figure 1.2. Shows a car entering a parking area. The car plate is recognized and stored and upon its exit the car plate is read again and the driver is charged for the duration of parking.

Figure 1.2 Car inside a parking area

- **Automatic Toll Gates:** Manual toll gates require the vehicle to stop and the driver to pay an appropriate tariff. In an automatic system the vehicle would no longer need to stop. As it passes the toll gate, it would be automatically classified in order to calculate the correct tariff. This is shown in Figure 1.3.
- **Border Crossing:** This application assists the registry of entry or exits to a country, and can be used to monitor the border crossings. This installation is shown in Figure 1.4. Each vehicle information is registered into a central database and can be linked to additional information.

Figure 1.3 Car at the toll booths

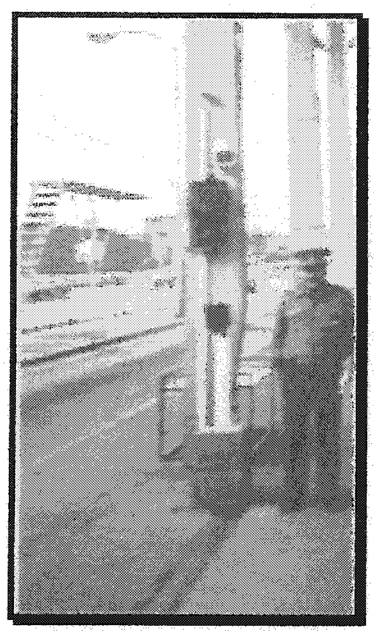

Figure 1.4 Installed LPR system at border crossing.

1.2 Elements of Typical LPR System

LPR systems normally consist of the following units:

- **Camera-** Takes image of a vehicle from either front or rear end.
- **Illumination-** A controlled light that can bright up the plate, and allows day and night operation. In most cases the illumination is Infra-Red (IR) which is invisible to the driver.
- **Frame Grabber-** An interface board between the camera and the PC that allows the software to read the image information.
- **Computer-** Normally a PC running Windows or Linux. It runs the LPR application that controls the system, reads the images, analyzes and identifies the plate, and interfaces with other applications and systems.
- **Software-** The application and the recognition package.
- **Hardware-** Various input/output boards used to interface the external world (such as control boards and networking boards).
- **Database-** The events are recorded on a local database or transmitted over the network. The data includes the recognition results and (optionally) the vehicle or driver-face image file.

1.3 Working of Typical LPR System

When the vehicle approaches the secured area, the LPR unit senses the car and activates the illumination (invisible infra-red in most cases) as shown in Figure 1.5a. The LPR unit takes the pictures from either the front or rear plates from the LPR camera. The image of the vehicle contains the license plate.

The LPR unit feeds the input image to the system. The system then enhances the image, detects the plate position, extracts the plate, segments the characters on the plate and recognizes the segmented characters.

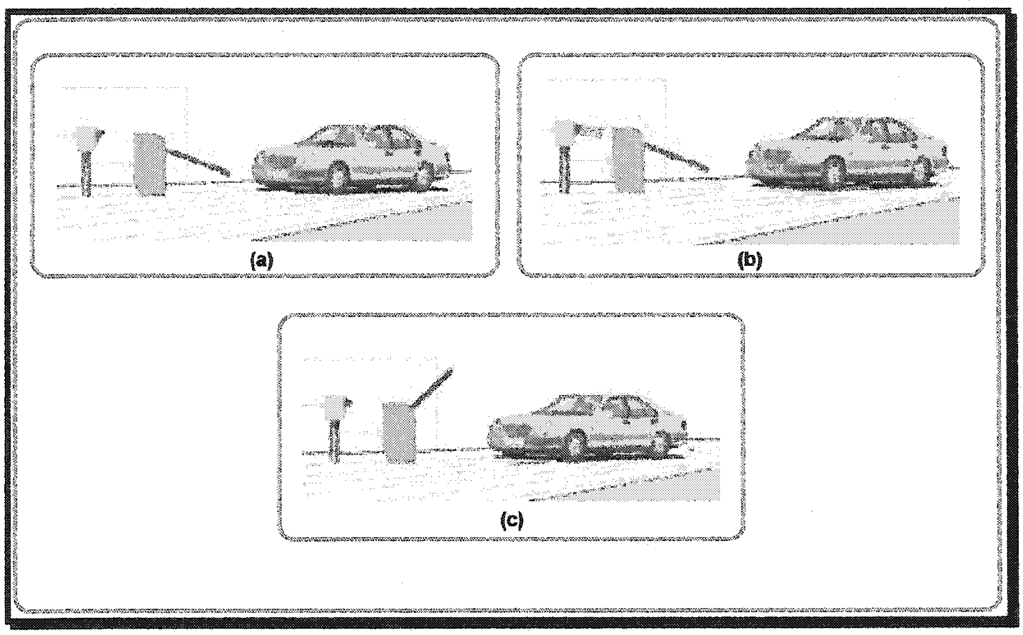

Figure 1.5 (a) Car approaching Secured Area, (b) Installed LPR System Sensing the Car, (c) Car entering after Recognition

The LPR unit then checks if the vehicle appears on a predefined list of authorized vehicles. if found, it signals to open the gate by activating its relay. The unit can also switch on a green "go-ahead" light or red "stop" light. The unit can also display a welcome message or a message with personalized data. The authorized vehicle enters into the secured area, as shown Figure 1.5c. After passing the gate its detector closes the gate. Now the system waits for the next vehicle to approach the secured area.

1.4 Structure of the Proposed System

The system presented is designed to recognize license plates from the front and rear of the vehicle. Input to the system is an image sequence acquired by a digital camera that consists of a license plate and its output is the recognition of characters on the license plate. The system consists of the standard four main modules in an LPR system, viz. Image acquisition, License plate extraction, License plate segmentation and License plate recognition. The structure of the system is shown in Figure 1.6. The first task acquires the selected portion of the image (i.e., the portion which contains a license plate). The second task extracts the region that contains the license plate. The third task isolates the six characters, consisting of three letters and three numerals, as in the case of Saudi Arabian License Plates. The last task identifies or recognizes the segmented characters.

1.4.1 Image Acquisition

This is the first phase in an LPR system. This phase deals with acquiring an image by an acquisition method. In our proposed system, we use a high resolution digital camera to acquire the input image. The input image is 640x480 pixels.

1.4.2 License Plate Extraction

License Plate Extraction is a key step in an LPR system, which influences the accuracy of the system significantly. This phase extracts the region of interest, i.e., the license plate, from the acquired image. The proposed approach involves four steps, viz., Vertical Edge Detection, Size-and-Shape Filtering, Vertical Edge Matching and Finding B/W (Black/White) ratio. These steps will be discussed in Section 4.2.

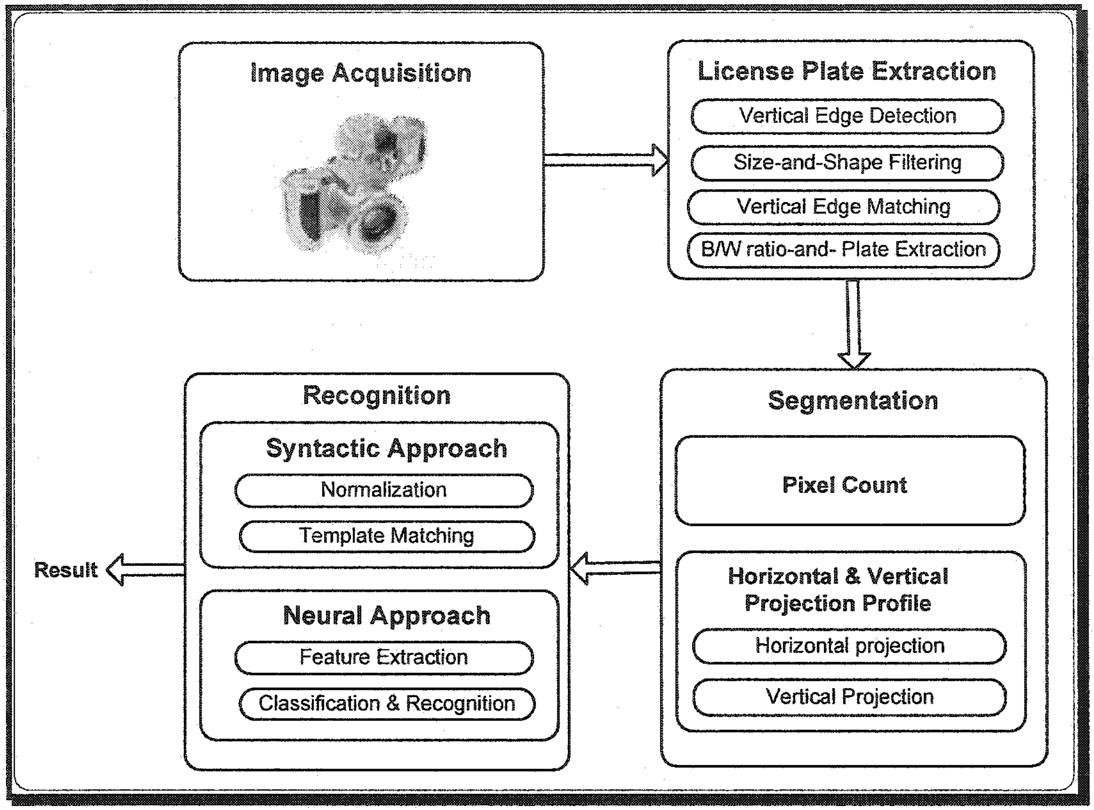

Figure 1.6 Structure of the Proposed System.

1.4.3 License Plate Segmentation

License Plate Segmentation, which is sometimes referred to as Character Isolation takes the region of interest and attempts to divide it into individual characters. To ease the process of detecting the characters, the extracted plate is divided into six images, each containing one isolated character. Since Saudi Arabian license plates consists of 6 characters, with 3 letters and 3 numerals. The proposed approach discusses segmentation using two methods, Pixel Count and Horizontal & Vertical Projection. This will be elaborated in Section 4.3.

1.4.4 License Plate Recognition

The last phase in LPR system is to recognize the isolated characters. After splitting the extracted license plate into six images, the character in each image can be identified. There are many methods used to recognize isolated characters. We have investigated recognition phase using Syntactic approach and Neural network approach. This is explained in more detail in Section 4.4

1.5 Objective

The work presented here aims at the following aspects.

- study the existing license plate recognition systems,
- develop a new technique or enhance existing techniques for each phase in a license plate recognition system,
- compare the various techniques at hand with the proposed system, and
- build a system that delivers optimal performance both in terms of speed and accuracy.

1.6 Thesis Organization

This thesis presents a complete system for the recognition of Saudi Arabian license plates. The thesis is structured as follows:

- **Chapter 2:** A review of the previous contributions to this work. A brief literature review of the techniques used in the four phases of an LPR system is presented, followed by an overview of the various commercial license plate recognition systems that have been developed.

- **Chapter 3:** A brief introduction to neural networks. It discusses in detail feed forward neural networks, and particularly, multilayer perceptron.
- **Chapter 4:** A conceptual model of our proposed LPR system is presented. It discusses various stages in the design and in particular the techniques used at various stages.
- **Chapter 5:** Experimental analysis and the results obtained at different stages under different conditions are discussed.
- **Chapter 6:** Conclusions about our work and future work are presented.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

License plate recognition systems have received a lot of attention from the research community. Much research has been done on Korean, Chinese, Dutch and English license plates. A distinctive feature of research work in this area is being restricted to a specific region, city, or country. This is due to the lack of standardization among different license plates (i.e., the dimension and the layout of the license plates). This section gives an overview of the research carried out so far in this area and the techniques employed in developing an LPR system in lieu of the following four stages: image acquisition, license plate extraction, license plate segmentation and license plate recognition phases. In the next section various existing or novel methods for the image acquisition phase are presented. Then, we present some previous work relevant to the license plate extraction phase. This is followed by some existing techniques in the segmentation of license plate into individual characters. After that, we review existing methods for recognition of individual Arabic characters. This chapter concludes by reviewing some of the commercially developed license plate recognition systems.

2.2 Image Acquisition

Image Acquisition is the first step in an LPR system and there are a number of ways to acquire images, the current literature discusses different image acquisition methods used by various authors. Yan et. al. [43] used an image acquisition card that converts video signals to digital images based on some hardware-based image preprocessing. Naito et. al. [28,29,30] developed a sensing system, which uses two CCDs (Charge Coupled Devices) and a prism to split an incident ray into two lights with different intensities. The main feature of this sensing system is that it covers wide illumination conditions from twilight to noon under sunshine, and this system is capable of capturing images of fast moving vehicles without blurring. Salgado et. al. [38] used a Sensor subsystem having a high resolution CCD camera supplemented with a number of new digital operation capabilities. Kim at. al. [23] uses a video camera to acquire the image. Cornelli et. al. [6] used a TV camera and a frame grabber card to acquire the image for the developed vehicle LPR system.

2.3 License Plate Extraction

License plate extraction is the most important phase in an LPR system. This section discusses some of the previous work done during the extraction phase Hontani et. al. [15] proposed a method for extracting characters without prior knowledge of their position and size in the image. The technique is based on scale shape analysis, which in turn is based on the assumption that, characters have line-type shapes locally and blob-type shapes globally. In the scale shape analysis, gaussian filters at various scales blur the given image and larger size shapes appear at larger scales. To detect these scales the idea of principal curvature plane is introduced. By means of normalized principal curvatures, characteristic points are extracted from the scale space x-y-t. The position (x, y) indicates the position of the figure and the scale *t* indicates the inherent characteristic size of corresponding figures. All these characteristic points enable the extraction of the figure from the given image that has line-type shapes locally and blob-type shapes globally. Kim et. al. [23] used two Neural Network-based filters and a post processor to combine two filtered images in order to locate the license plates. The two Neural Networks used are vertical and horizontal filters, which examine small windows of vertical and horizontal cross sections of an image and decide whether each window contains a license plate. Cross-sections have sufficient information for distinguishing a plate from the background. Lee et. al. [25] and Park et. al. [34] devised a method to extract Korean license plate depending on the color of the plate. A Korean license plate is composed of two different colors, one for characters and other for background and depending on this they are divided into three categories. In this method a neural network is used for extracting color of a pixel by HLS (Hue, Lightness and Saturation) values of eight neighboring pixels and a node of maximum value is chosen as a representative color. After every pixel of input image is converted into one of the four groups, horizontal and vertical histogram of white, red and green (i.e. Korean plates contains white, red and green colors) are calculated to extract a plate region. To select a probable plate region horizontal to vertical ratio of plate is used. Dong et. al [5] presented histogram based approach for the extraction phase. Kim G. M [22] used Hough transform for the extraction of the license plate. The algorithm behind the method consists of five steps. The first step is to threshold the gray scale source image, which leads to a binary image. Then in the second stage the resulting image is passed through two parallel sequences, in order to extract horizontal and vertical line segments respectively. The result is an image with edges highlighted. In the third step the resultant image is then used as input to the Hough transform, this produces a list of lines in the form of accumulator cells. In fourth step, the above cells are then analyzed and line segments are computed. Finally the list of horizontal and vertical line segments is combined and any rectangular regions matching the dimensions of a license plate are kept as candidate regions. The disadvantage is that, this method requires huge memory and is computationally expensive.

2.4 Segmentation

This section discusses previous work done for the segmentation of characters. Many different approaches have been proposed in the literature and some of them are as follows, Nieuwoudt et. al. [31] used region growing for segmentation of characters. The basic idea behind region growing is to identify one or more criteria that are characteristic for the desired region. After establishing the criteria, the image is searched for any pixels that fulfill the requirements. Whenever such a pixel is encountered, its neighbors are checked, and if any of the neighbors also match the criteria, both the pixels are considered as belonging to the same region. Morel et. al. [27] used partial differential equations (PDE) based technique , neural network and fuzzy logic were adopted in [32] for segmentation into individual characters.

Hansen et. al [12] uses the connected component method to segment the characters. This method works on binary images using simple morphology functions. The method for extracting connected components in a binary image is to search for a black pixel. When such a pixel is found, it is assumed that it is a part of a component and therefore forms the basis for an iterative process, based on Equation below:

$$X_k = (X_{k-1} \oplus B) \cap A, \qquad k = 1,2,3.....$$

where X_k represents the extracted component, A is the source image and B is a structuring element of size 3x 3 indicating 8-connectivity neighboring. The advantage of this method is that it is independent of the image rotation and that the bounds found are very precise. Its disadvantage is that it requires a good image quality and a good conversion from the original image to the binary image, to avoid making two or more characters appear as one connected region. Small gaps within the characters can also cause the method to fail.

2.5 Recognition

This section presents the methods that were used to classify and then recognize the individual characters. The classification is based on the extracted features. These features are then classified using either the statistical, syntactic or neural approaches. Some of the previous work in the classification and recognition of characters is as follows, Hasen et. al. [12] discusses a statistical pattern recognition approach for recognition but their technique found to be inefficient. This approach is based on the probabilistic model and uses statistical pattern recognition approach.

Cowell et. al. [7] discussed the recognition of individual Arabic and Latin characters. Their approach identifies the characters based on the number of black pixel rows and columns of the character and comparison of those values to a set of templates or signatures in the database. Cowell et. al. [8] discusses the thinning of Arabic characters to extract essential structural information of each character which may be later used for the classification stage. Mei Yu et. al. [45] and Naito et. al. [29] used template matching. Template matching involves the use of a database of characters or templates. There is a separate template for each possible input character. Recognition is achieved by comparing the current input character to each of template in order to find the one which matches the best. If $I(x,y)$ is the input character, $T_n(x,y)$ is template n, then the matching function $s(I,T_n)$ will return a value indicating how well template n matches the input character (Equation 2.1). Some of the common matching functions are:

City block $$s(I,T_n)=\sum_{i=0}^{w}\sum_{j=0}^{h}|I(i,j)-T_n(i,j)| \quad (2.1)$$

Euclidean $$s(I,T_n)=\sum_{i=0}^{w}\sum_{j=0}^{h}(I(i,j)-T_n(i,j))^2 \quad (2.2)$$

Character recognition is achieved by identifying which T_n gives the best value of matching function, $s(I,T_n)$. The method can only be successful if the input character and the stored templates are of the same (or at least very similar) font. Template matching can be performed on binary, thresholded characters or on gray-level characters. Bokser [4] uses zoning on solid binary characters. In this method an *nxm* grid is superimposed on character image, and for each nxm zones, the average gray level is computed. The obtained average gray levels are used as features for the recognition of characters. Hamami et. al. [11] adopted a structural or syntactic approach to recognize characters in a text document, this technique can yield a better result when applied on the recognition of individual characters. This approach is based on the detection of holes and concavities in the four directions (up, down, left and right), which permits the classification of characters into different classes. In addition, secondary characteristics are used in order to differentiate between the characters of each class. The approaches discussed in this paragraph are based on the structural information of the characters and uses syntactic pattern recognition approach.

Hu [16] proposed seven moment that can be used as features to classify the characters. These moments are invariant to scaling, rotation and translation. The obtained moments acts as the features, which are passed to the neural network for the classification or recognition of characters. Zernike moments have also been used by several authors [3,20,21] for recognition of characters. Using zernike moments both the rotation variant and rotation invariant features can be extracted. These features then uses neural network for the recognition phase. Neural network accepts any set of distinguishable features of a pattern as input. It then trains the network using the input data and the training algorithms to recognize the input pattern (In this case characters . The detailed discussion about neural network is provided in chapter 3.

2.6 Commercial Products

2.6.1 IMPS (Integrated Multi-Pass System)

An IMP [33] is a Singaporean commercially developed license plate recognition system. It is a high performing robust system that gives consistent results under all weather conditions. Using advanced image processing and artificial intelligent techniques such as AI best first breadth-wise search algorithm, combined template and neural network recognizers, fuzzy logic and an arsenal of image processing tools, it automatically locates vehicle license plates and reads the numbers accurately each time every time.

2.6.2 Perceptics

Perceptics [36] is the world leader in license plate reader technology. Current LPR system read Latin (A-Z) and Korean (Hangul) letter and Arabic number (0-9); however, the LPR can be programmed to read any language or symbol in any alphanumeric combination or context on both retro and non-retro reflective plates. With milliseconds the LPR system locates, captures and identifies a vehicle's license plate data and makes a read decision. The system's reliability and flexibility allow it to accommodate some of the most stringent needs in some of the worst conditions. Features of this LPR technology include:

- Automatic and within milliseconds
- Reads accurately in most weather conditions
- Reads accurately at highway speeds.
- Works 24 hours a day, 7 days a weeks.

2.6.3 Vehicle Identification System for Parking Areas (VISPA)

PPI's Vehicle Identification System for Parking Areas (VISPA) [35], uses video imaging for better recognition, identification and improved security. VISPA provides for state-of-the-art video technology, easy installation and has accessories and features for most parking security surveillance needs.

Features:

- Open architecture to most common video-systems.
- Compatible with standard hardware and software.
- Can be customized according to specific user needs.

VISPA is available in two forms:

Basic Version

Image Recognition: An image of the car and/or the driver (depending on the location of your camera) will be taken as soon as the car approaches the triggering device. The image will be linked to the ticket. The basic system version connects to 4 cameras and can be upgraded to 8 cameras.

Enhanced Version

License Plate Identification: The VISPA controller with an integrated frame grabber card for 4, 8, or 16 cameras automatically identifies the license plate from the video image and stores it in a database. The license plate can then be encoded on the ticket.

2.6.4 Zamir Recognition System Ltd. (ZAMIR)

Zamir [46] provides a fully integrated LPR system including *imaging units*, a DPU containing LPR ware *Zamir's LPR engine* and a full platform for *access control management*.

Zamir's Imaging Unit: Zamir uses its electro optic capabilities to design and manufacture a unique imaging unit. This unit, positioned by the traffic lane or gate will create a special image needed for the LPR engine to achieve best recognition results. The imaging unit is designed to produce the same image under all ambient light and weather conditions.

Zamir's LPR Engine: The heart of the system is the recognition engine. Although LPR technology is font, shape and color sensitive, Zamir developed a learning mode that allows the engine to learn almost any new license plate type with minimal resources. Zamir's LPR engine is constantly being updated to recognize license plates from various countries.

Insignia – Access control management platform: Zamir developed a full access control and security management software module, called the InSignia. InSignia includes a user database, black list database, gate and lot tables, grouping and hierarchical possibilities and an event journal. The program also includes a simple report generator. This full set of tools allows you to create almost any grouping and limitation possibilities, manage many access points from one central point over a network, update and control records over the web etc.

2.6.5 Hi-Tech Solution

Hi-Tech Solutions [13] is a system and software company that develops cutting edge optical character recognition (OCR) solutions by implementing the company's unique image processing software and hardware in a wide range of security and transportation applications. There technology is based on computer vision, the system read the camera images and extract the identification data from the images. The recognition result is then logged together with the images. This is the main advantage of vision based recognition, the records include both the image plus the extracted result. There product includes,

SeeCar License Plate Recognition: Detects and reads Vehicle license plates for parking, access control, traffic surveillance, law enforcement and security applications. Available as a complete system which is based on a background Windows application, Windows

DLL or Linux library, as a stand-alone turn-key version, or in form of different special-task systems.

SeeContainer Identification System: Tracks and reads Shipping container identification marking, and transmits the ID string to the port or gate computer, or to a client process. Available as complete systems, such as SeeGate - a recognition system for the Trucks and Containers, or SeeCrane - crane mounted Container recognition system.

2.7 Summary

This chapter reviewed material relevant to the license plate recognition system. The relevant techniques used in the four phases of an LPR system were discussed. Several commercially available and developed LPR systems is also presented. In the case of image acquisition, a sensing system using two Charge Coupled Devices along with a prism gives better input to the system. Because the main feature of this sensing system is that it covers wide illumination conditions from twilight to noon under sunshine, and this system is capable of capturing images of fast moving vehicles without blurring video camera with a frame. In the case of license plate extraction, Hough transform was used to extract the license plate by using storing the horizontal and vertical edge information. But the disadvantage is that, this method requires huge memory and is computationally expensive. In the proposed system, this method was enhanced and improved by just considering the vertical edges. The details of the proposed approach are discussed in Chapter 4. Various segmentation techniques were presented in the segmentation stage. Then the literature for recognition of characters using various approaches was also discussed. Lastly, some of the number plate recognition systems which have been developed commercially were presented.

CHAPTER 3

ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks (ANNs), which are also referred to as neural computation, connectionist models, and parallel distributed processing (PDP), are massively parallel computing systems consisting of an extremely large number of simple processors with many interconnections between them. ANNs were designed with the goal of building "intelligent machines" to solve complex problems, such as pattern recognition and optimization, by mimicking the network of real neurons in the human brain [19].

This chapter gives a basic introduction to neural networks. Section 3.1 describes the network architecture and presents taxonomy of the various neural network architectures. The most common neural network architecture available for an character recognition application is the feed forward neural network. Section 3.2 explains the feed forward network and the two types of feed forward networks: The Multi Layer Perceptron and the Radial Basis Function. Finally in section 3.3 we compare the two types of Neural Networks.

3.1 Network Architecture

Neural Network classifiers determine which of M classes is most representative of an unknown static input pattern containing N input elements. In an image classifier the inputs might be the gray scale level of each pixel for a picture and the classes might represent different objects. Unlike the traditional classifier, in which strong assumptions are typically made concerning underlying distributions of the input elements, the neural net makes no assumptions concerning the shape of underlying distributions but focuses on errors that occur where distributions overlap. It may thus be more robust than classical technique and works well when inputs are generated by non linear processes and are heavily skewed.

An assembly of artificial neurons is called an artificial neural network. ANNs can be viewed as weighted directed graphs in which nodes are artificial neurons and directed edges (with weights) are connections from the outputs of neurons to the inputs of neurons. Based on the connection pattern (architecture), various ANNs can be grouped into two major categories as shown in Figure 3.1(a) ***(Feed Forward Neural Networks)*** in which no loop exists in the graph, and Figure 3.1(b) ***(Feed Back or Recurrent Neural Networks)*** in which loops exist because of feedback connections. The most common family of feed forward networks is a layered network, in which neurons are organized into layers with connections strictly in one direction from one layer to another. In fact, all the networks with no loops can be rearranged in the form of layered feed forward networks with possible skip-layer connections. Figure 3.1 also shows typical networks of each category. Generally speaking, feed forward networks are static networks, i.e., given an input, they produce only one set of output values, not a sequence of values. These networks are memory-less, in the sense that the response of a feed forward network to an input is independent of the previous state of the network. An exception is the time delay feed forward network in which dynamics occurs because of different delay factors of the neurons in the network.

Recurrent networks are dynamic systems. Upon presenting a new input pattern, the outputs of the neurons are computed. Because of the feed back paths, the inputs to each neuron are then modified, which leads the network to enter a new state. This process is repeated until convergence. Obviously, different mathematical tools must be employed to treat these two different types of networks. Dynamic systems are often described by differential equations. In this thesis we concentrate on the feed forward neural network. The following sections give a brief overview of the feed forward neural network. Following this the two types of neural networks, the Feed Forward Neural Network and the Feed Backward Neural Network will be presented.

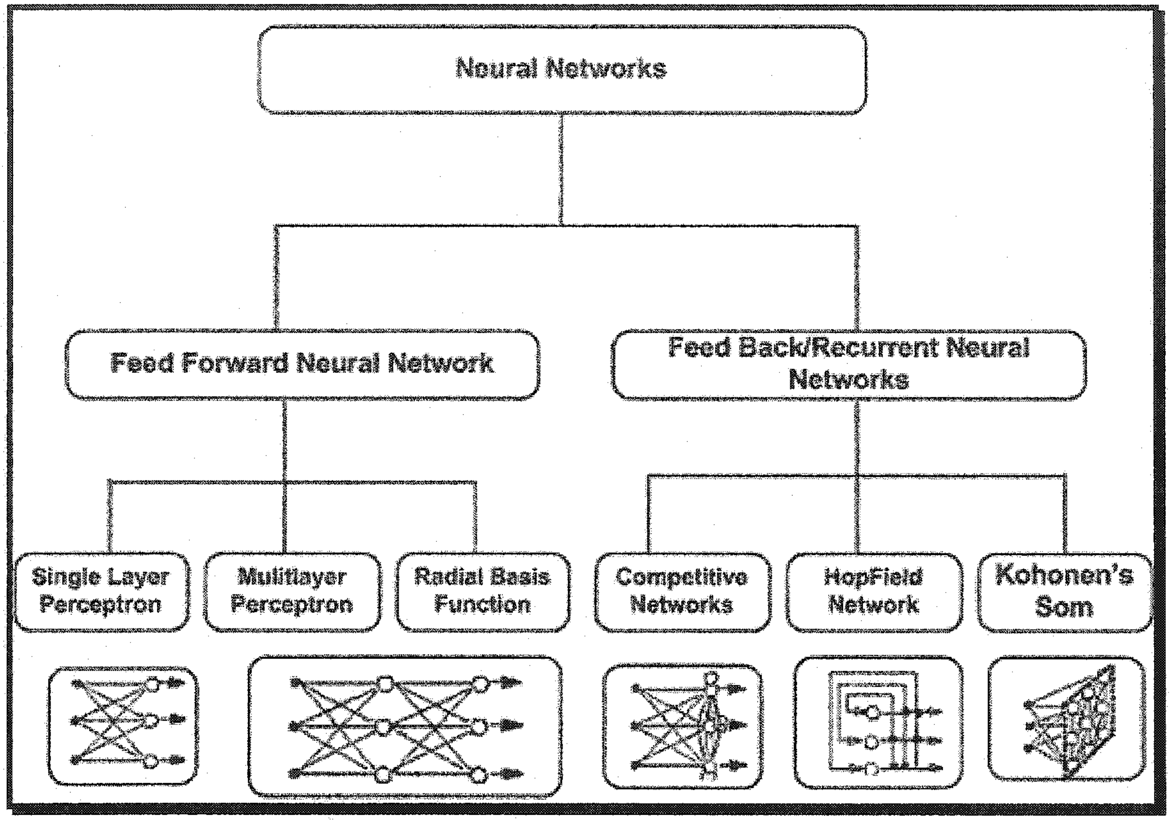

Figure 3.1 Taxonomy of Different Neural Network Architectures

3.2 Feed Forward Neural Network

The Perceptron is the simplest form of feed forward neural network that is said to be ***linearly Separable.*** In its simplest form, the perceptron, forms two decision regions that are separated by a hyper plane. Basically it consists of a single neuron with adjustable synaptic weights and bias. The single node computes a weighted sum of the input elements, subtracts a threshold (θ) and passes the result through a hard limiting nonlinearity such that the output is either +1 or -1.The decision rule is to respond to class A, if the output is +1 and class B, if the output is -1.

Rosenblatt [37]developed the first perceptron convergence procedure for adjusting weights. First the weights and thresholds are initialized to small random non zero values Then a input with N continuous valued elements is applied to the input and the output is computed. Connection weights are adapted only when an error occurs.

Rosenblatt [37] proved that if the inputs presented from the two classes are separable (That is if they fall on the opposite sides of the hyper plane) then the procedure converges and positions the decision hyper plane between the two classes. This decision hyper plane separates all the samples belonging to the class A from the samples belonging to class B. Figure 3.2 shows a single layer perceptron which determines whether a sample belongs to class A or class B.

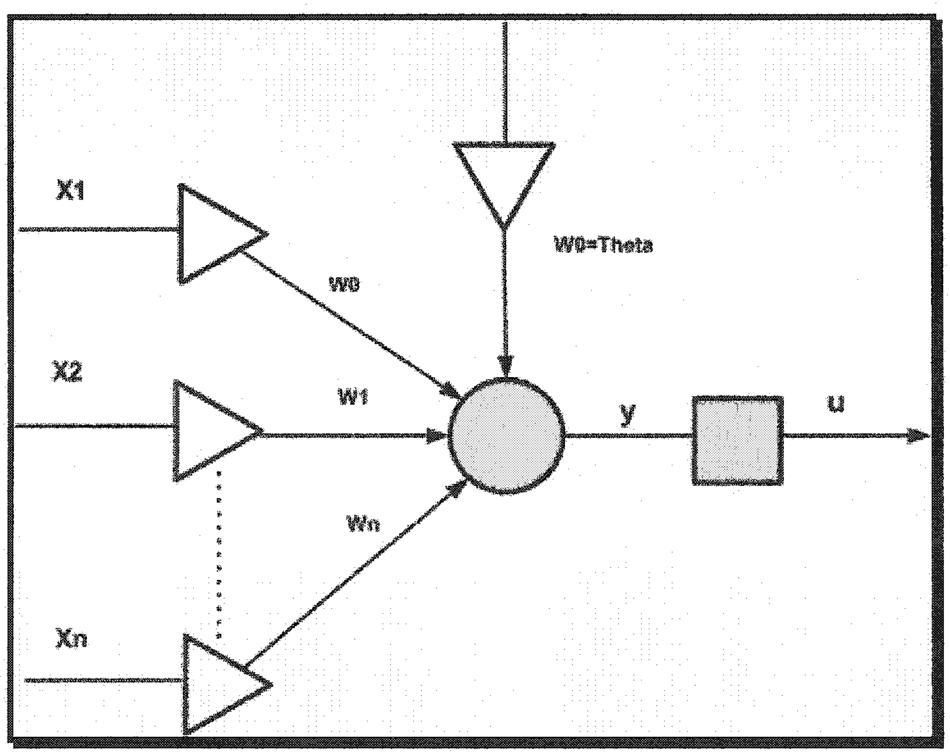

Figure 3.2: A single Layer Perceptron

One problem with the perceptron learning algorithm is that decision boundaries may oscillate continuously when inputs are not separable and distributions overlap. The least mean square (LMS) is a modification to the perceptron convergence procedure which minimizes the mean square error between the desired output of a perceptron like net and the actual output [14,42].

The LMS Algorithm is identical to the perceptron convergence algorithm except that the hard limiting non linearity is replaced by a threshold logic non linearity. Weights are corrected on every trial by an amount that depends on the difference between the desired and the actual input. A classifier that uses the LMS training algorithm could use the desired output of 1 for class A and 0 for Class B. During the operation the input would then be assigned to the class A only if the output was above 0.5.

3.2.1 Multilayer Perceptron

When classes cannot be separated by a hyper plane, the perceptron convergence procedure is not appropriate. Distributions for two classes for the exclusive OR problem is disjoint and cannot be separated by a single straight line. So the XOR Problem cannot be solved using a single Layer Perceptron. Multi Layer perceptrons overcome many of the limitations of the single layer perceptrons.

A multilayer perceptron consist of a set of sensory units (Source nodes) that constitute the input layer, one or more hidden layers of computation nodes and an output layer. Input signal propagates through the network in the forward direction, on a layer by layer basis.

Multilayer perceptrons have been successfully applied to solve some difficult and diverse problems by training them in a supervised manner with a highly popular algorithm known as the *error back – propagation algorithm.* This algorithm is based on the *error correction learning rule.*

Basically, error back propagation algorithm consists of two passes through the different layers of the network: A forward pass and a back ward pass. In the forward pass a ***Activity pattern*** (Input vector) is applied to the sensory nodes of the network, and its affect propagates through the network layer by layer. Finally a set of outputs is produced as the actual response of the network. During the *Forward Pass,* the synaptic weights of the network are all fixed. During the *Backward pass,* the synaptic weights of the network are adjusted in accordance with the error correction rule. Specially the actual response of the network is subtracted from the desired response to produce an *Error Signal.* This error signal is back propagated through the network against the direction of synaptic connections. Hence the algorithm is known as the "error back propagation algorithm". The learning process performed with the algorithm is called the Back propagation learning.

Figure 3.3 shows the architectural graph of a multilayer perceptron with two hidden layers and an output layer. The network shown in the figure is fully connected. This means that a neuron in any layer of the network is connected to all the nodes/neurons in the previous layer. Signal flow through the network progresses in the forward direction, from left to right and on layer by layer basis.

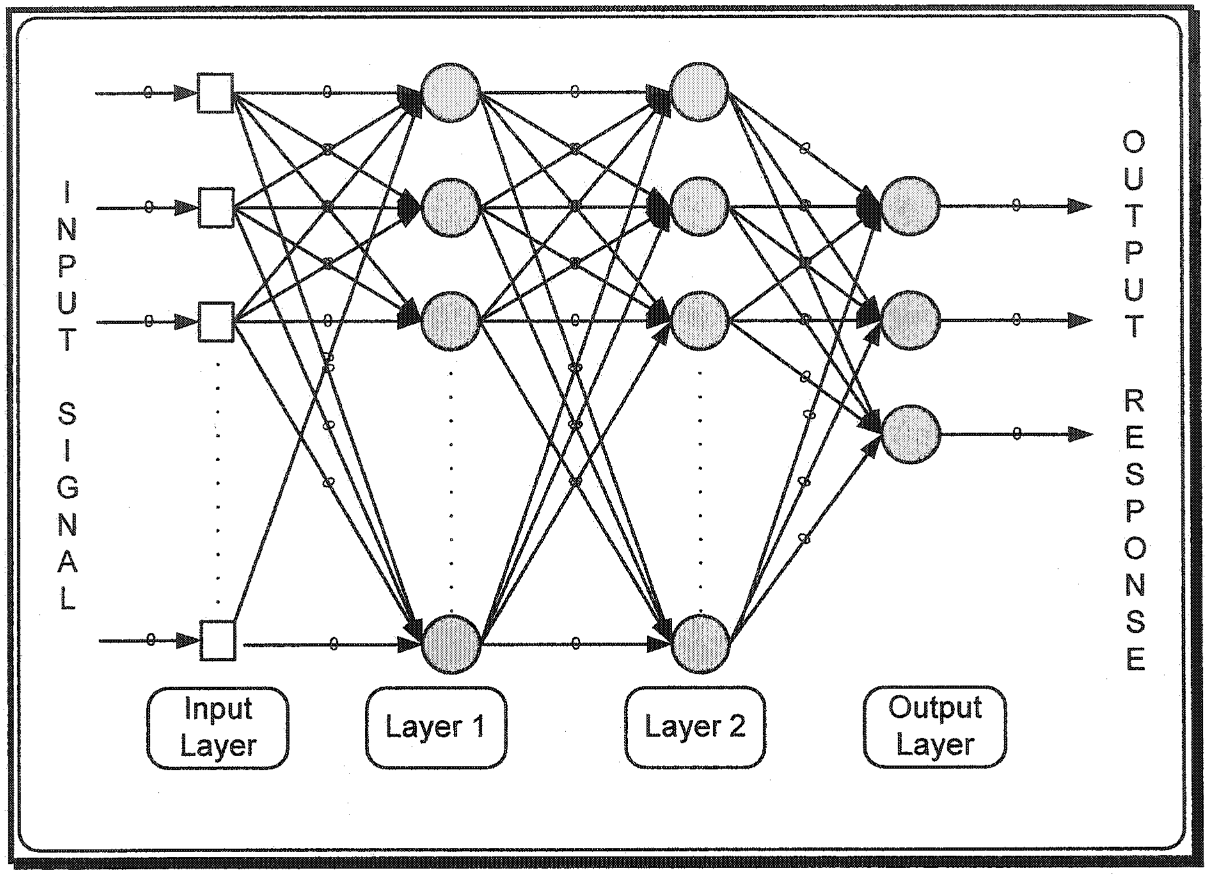

Figure 3.3 Architectural graph of a multilayer perceptron with two hidden layers.

3.2.2 Radial Basis Function Network

The second type of feed forward neural network is the Radial Basis Function Neural Network. The construction of a Radial – Basis Function (RBF) Network, involves three different layers. The input layer is made up of source nodes (Sensory Units) that connect the network to its environment. The second layer is the single hidden layer in the network that applies a non linear transformation from the input space to the hidden space. In most applications the hidden space is of high dimensionality. The output layer is linear supplying the response of the network to the activation pattern applied to the input layer.

When a Radial Basis Function network is used to perform a classification task, the problem is basically solved by transforming it into a high dimensional space in a non linear manner. Radial Basis Functions were first introduced in the solution of the real multivariate interpolation problem.

3.3 Comparison of RBF Networks and Multilayer Perceptron

Radial Basis Functions and Multilayer perceptron are both examples of layered feed forward networks. They are both universal approximators. However these two networks differ from each other from several different aspects.

- RBF has a single hidden layer, where as an MLP has one or more hidden layers.
- Typically the computation nodes of an MLP, located in a hidden or an output layer share a common neuron model. On the other hand, the computation nodes in the hidden layer of an RBF Network are quite different and serve a different purpose from those in the output layer of the network.
- The hidden layer of an RBF Network is non linear where as the output layer is linear. However the hidden and output layer of an MLP used as a pattern classifier is usually all nonlinear.
- The argument of the activation function of each hidden unit in an RBF Network computes the *Euclidean Norm (Distance)* between the input vector and the center of that unit. Mean while, the activation function of each hidden unit in an MLP

computes the inner product of the input vector and the synaptic weight vector of that unit.

3.4 Application of Neural Networks to Character Recognition

One of the important applications of Artificial Neural Networks is pattern classification. Various ANN models and learning algorithms have been successfully applied to a variety of pattern classification and recognition problems. Research shows that ANN's have proved to be quite successful for character recognition applications. However there is no conclusive evidence about ANN's superiority over conventional statistical pattern classifiers.

At the first census Optical Character Recognition System Conference in 1992, more than 40 different hand written character recognition systems were tested on the database. The top ten performers among them used either some type of feed forward neural networks or nearest neighbor classifier. ANNs tend to be superior in speed and memory compared to the statistical methods. One conclusion drawn from the test is that the recognition performance of character recognition systems is comparable to the human performance on isolated characters.

Many neural network architectures have been used in LPR implementations. The most popular neural networks are the multilayer feed forward neural network, where neurons are grouped between layers and connections between neurons in consecutive layers are permitted. Feed forward neural networks give the choice of Multilayer perceptron (MLP) or the Radial Basis Function Network (RBF). In this thesis we have used multilayer perceptron for the license plate recognition problem.

3.5 Summary

This chapter gave a brief introduction to the neural network. First, an overview of the network architecture of a neural network was presented. Following this a brief discussion regarding the feed forward neural network, especially, Multi Layer Perceptron and Radial Basis Functions were discussed. The next section gives a comparison between MLP and RBF networks. Lastly some of the applications of neural network to character recognition were discussed.

CHAPTER 4

CONCEPTUAL MODEL

A conceptual model for any pattern recognition system can be generally divided into a number of distinct stages: Data Collection, Input Reduction, Segmentation, Normalization, Feature Extraction and Classification. The goal of the overall process is to correctly classify the pattern being analyzed. Each of the stages has its unique goals that enhance that possibility.

- **Data Collection:** To accurately record raw data while minimizing the quantization noise.
- **Sampling:** To decrease the size of input data with a resultant decrease in complexity for training while minimizing loss of accuracy.

- **Segmentation:** To divide or separate data into defined, clearly understood blocks or segments.
- **Normalization:** To make the input invariant to scaling, rotation and translation.
- **Feature Extraction:** To further reduce the input space by grouping inputs into relevant features.
- **Classification:** To correctly classify the input as one of the output classes.

In the conceptual model for our system, data is collected by acquiring the image of a vehicle. Then we process the input data and reduce or sample it such that, we get the region of interest (i.e., the license plate). Followed the extraction phase, is the segmentation of extracted plate into individual characters. In the final stage, first the features of the segmented characters are extracted and then these features are given to the classification stage to get the recognized output.

This chapter is organized in the following way. The first section deals with collection of data by acquiring images of the vehicle using a digital camera. The second section discuses the extraction of license plate. The extraction process is further divided into four main stages,viz, vertical edge detection, size-and-shape filtering, vertical edge matching and finding the black-to-white ratio. The next section presents the segmentation of the extracted plate into individual characters. Followed by the segmentation stage is the recognition of the segmented characters, which is discussed both using the syntactic and neural approaches. Lastly, we conclude by giving a brief summary of the whole chapter.

4.1 Image Acquisition

This is the first phase in the LPR system. There are basically following three ways of acquiring an image. They are:

- ❖ Using a conventional analog camera and a scanner.
- ❖ Using a digital camera.
- ❖ Using a video camera and a frame grabber (capture card) to select a frame.

The first method, using a conventional analog camera, is clearly inappropriate for the LPR system since it is time consuming, tedious, and impractical. The second method i.e., using a digital camera is more practical, cost effective and reliable. The third one uses a video camera with a frame grabber that could be used in real life system to make the system automated, and is suitable for real time processing.

In the proposed system a high resolution digital camera is used to acquire the image. The acquired image as shown in Figure 4.1, is first converted to gray scaled image. Conversion to gray scale facilitates the extraction of the license plate. The gray-scaled image is shown in Figure 4.2.

4.2 License Plate Extraction

License plate extraction is the key step in an LPR system, which influences the accuracy of the system significantly. The goal of this phase is, given an input image, to produce a number of candidate regions, with high probability of containing a license plate. In the adopted approach, extraction of the license plate is divided into four steps which are explained in the following subsections.

4.2.1 Vertical Edge Detection

For the transformed gray-scaled image, its corresponding vertical edges are detected using Sobel or Prewitt edge detectors. The proposed system uses Sobel edge detector because it gives better results. The threshold used by the edge detector is dynamic because the system takes an automatic value from the algorithm. The Sobel edge detector uses a 3x3 mask, which is applied on the input image to give the resultant edged image. The edge detection algorithm is not time consuming since the algorithm used, is a built-in feature of

Figure 4.1 Original Image

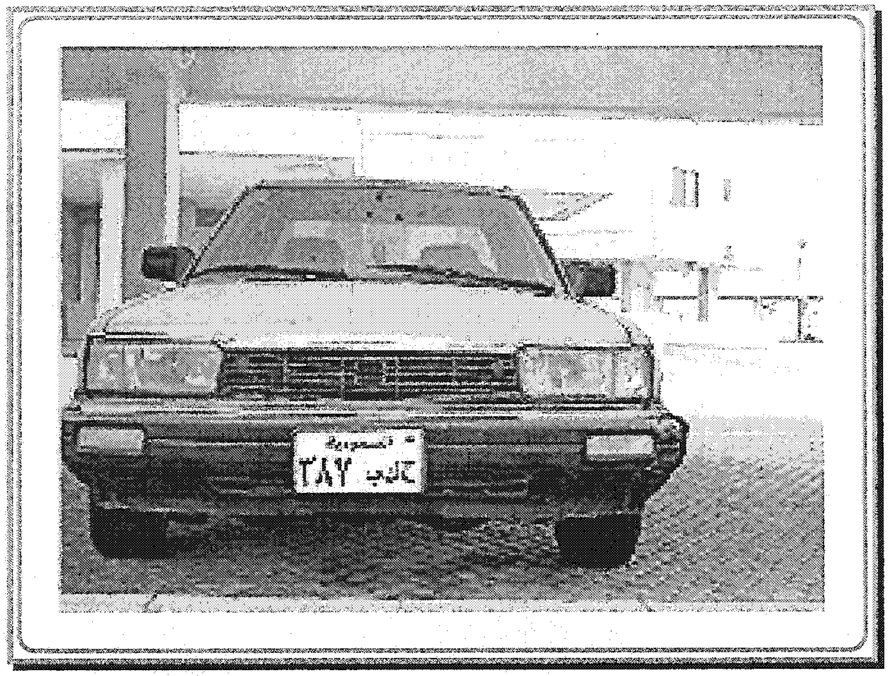

Figure 4.2 Gray Scaled Image

MATLAB6.1 [26]. Figure 4.3 shows an image with vertical edges. It is observed that most of the vehicles usually have more horizontal lines than vertical lines [45]. To reduce the complexity of algorithm, the vertical edges are detected. If two of the vertical edges are detected correctly, four corners of the license plate can then be located. This helps in extracting the license plate exactly from the input image even if it is out of shape.

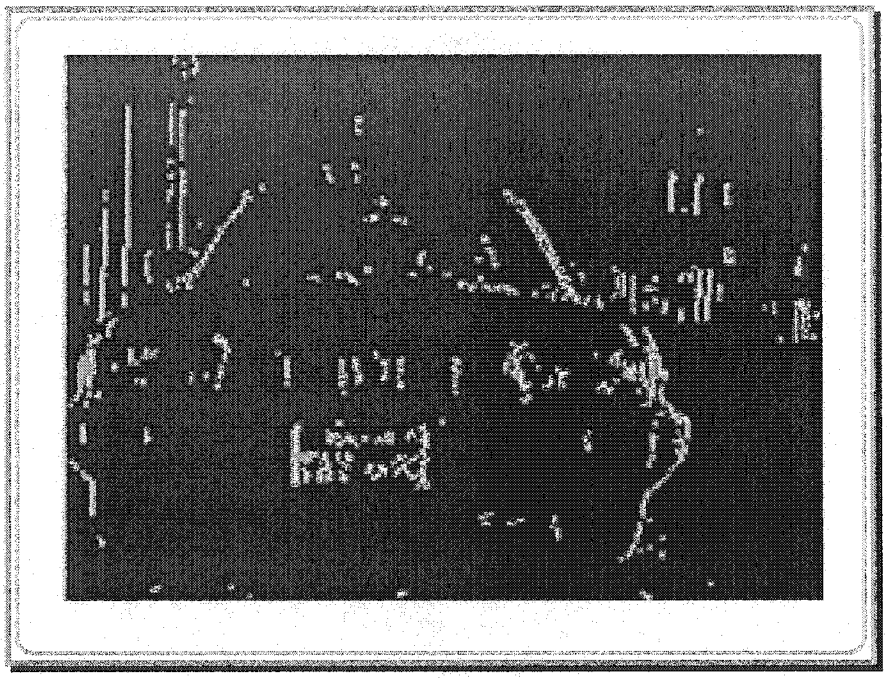

Figure 4.3 Vertical Edge Image

Figure 4.4 Vertical Edge Regions

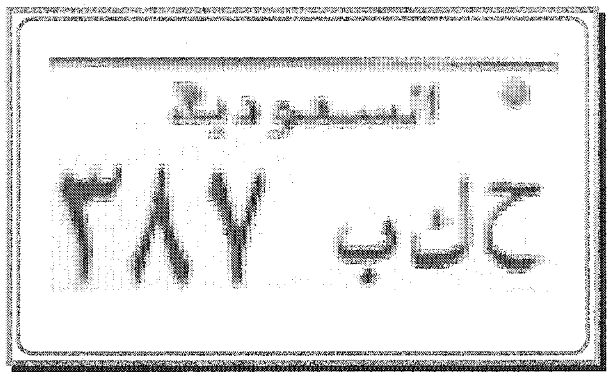

Figure 4.5 Extracted License Plate

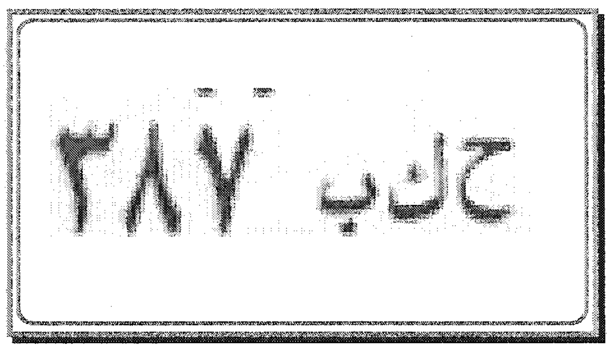

Figure 4.6 Cropped License Plate

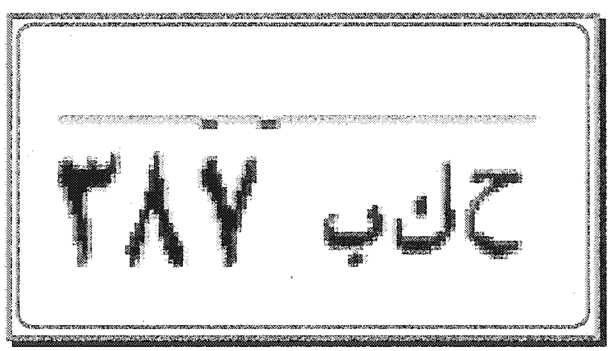

Figure 4.7 Cropped Binary License Plate

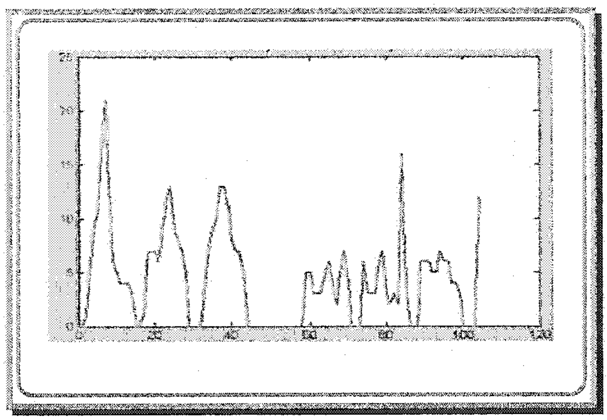

Figure 4.8 Histogram Depicting Vertical Projection on Figure 4.7

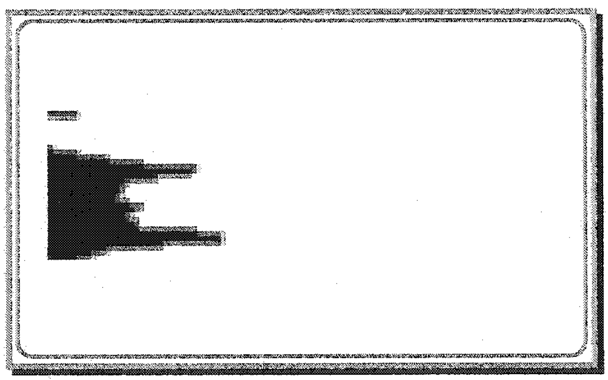

Figure 4.9 Horizontal Projection Profile of Figure 4.7

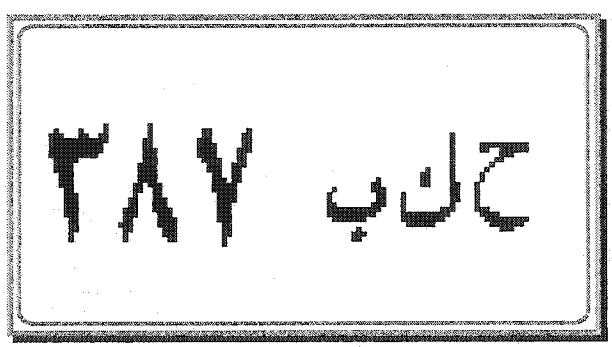

Figure 4.10 Resultant cropped binary license plate after horizontal projection

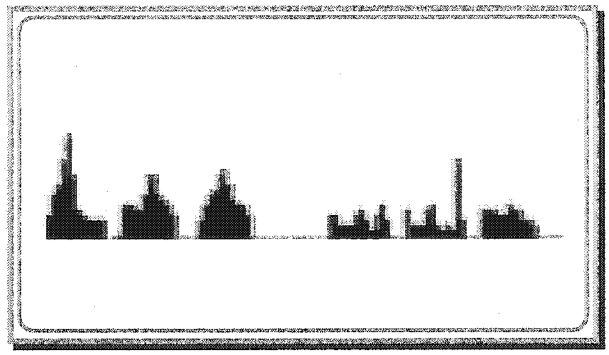

Figure 4.11 Vertical Projection Profile

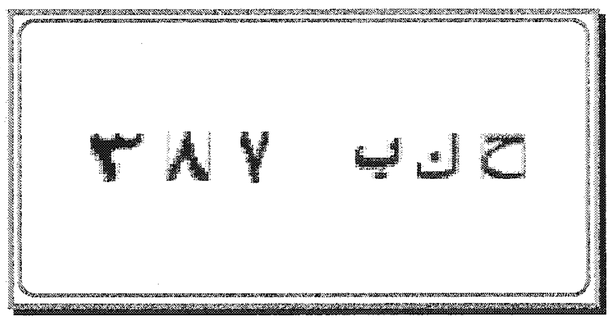

Figure 4.12 Segmented Characters

4.2.2 Size-and-Shape Filtering

Filtering is basically used to remove objects that do not satisfy some specific features. In the proposed approach, seed-filling algorithm by smith [40] is used to filter the unwanted objects. After filtering, the result is shown in Figure 4.4. The process of filtering starts by first detecting all the regions. Any group of white pixels is called a region if they are eight-connected pixels. In Figure 4.13., the eight pixels around the shaded pixel are called eight-connected components of the shaded pixel. For the proposed system, all the eight components are not considered. Only the components to the East, South-East, South and South-West (filled with blue, as shown in Figure 4.13) of the shaded pixel are considered.

If any of these four pixels are marked as a white pixel, it is taken as the next pixel to be processed, and the pixel under consideration is marked as the processed pixel.

Through filtering our target is to select the regions that can serve as possible license plate boundaries and discard the others by filling it with black pixel. Then we apply size and shape filter to get the region of interest (those regions that contain the license plate). To achieve this target any region of interest must be a straight line of specific length. So for each region the size and shape filter is applied. The first pixel of the region is marked as P_o and the last is marked as P_{end}.

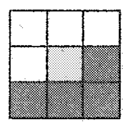

Figure 4.13: Eight-connected components

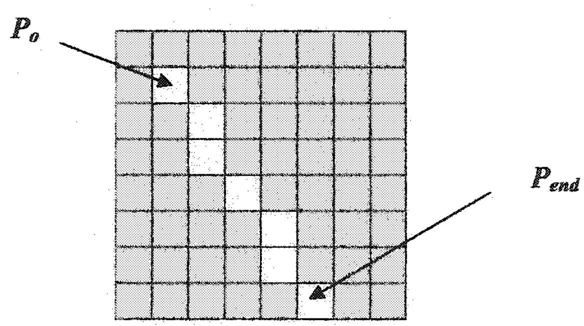

Figure 4.14: Example of an eight-connected region

An overall filtering algorithm is presented in Table 4.1

Table 4.1 Filling Algorithm

1. For each row
2. Traverse from first column to last column (West to East)
3. if a white pixel is encountered and is not marked as part of any other region
    a. Select the pixel as P_o
    b. Select the white pixels on East,SW,South and SE recursively until a white pixel having no eight-connected white pixel is found.
    c. Mark the last white pixel as P_{end}.
    d. if the slope and distance of line P_o,P_{end} falls in desired range
        i. mark all the pixels in the region as they are a part of a selected region.
    e. else
        i. fill all the pixels in the current region with black color, i.e., discard them.
4. end for

Shape of the region is detected through the slope of the line joining P_o and P_{end}. The size of the region is the distance between P_o and P_{end}. If the distance falls within the thresholded value, it is taken as the region of interest. A threshold of 30 to 100 pixels is taken as the distance between P_o and P_{end} with an angle of deviation of around ±15°. This threshold is taken since the image can be taken from a closer or farther view with some angle of deviation. The region in Figure 4.14 is discarded on the basis of slope of the line P_o, P_{end}.

4.2.3 Vertical Edge Matching

In this phase the width to height ratio of license plate is used to match the vertical edges tor find the region where there is a high probability of license plate. The standard ratio of width to height of Saudi Arabian license plate is about 2:1. After the size-and-shape filtering, the image in Figure 4.4 is extracted with a number of vertical regions. Only two of the regions could be the possible boundary of the license plate. The task of vertical edge matching is to find out the correct pair of regions that include the license plate. To achieve this task, the width-to-height ratio of the rectangular area between two vertical regions is compared with the actual standard ratio of a license plate. The standard ratio is taken with certain threshold to lie between 1.75 : 1 to 2.25 : 1 and the angle of deviation is ±15 with respect to the straight line perpendicular to x-axis. Figure 4.5 shows the extracted license plate for the matched vertical edges.

An algorithm for vertical edge matching is as shown in Table 4.2

Table 4.2 Edge Matching Algorithm

1. for i=1 to no. of extracted regions
2. for j=i+1 to no. of extracted regions
    a. if width-to-height ratio of region (i) and region (j) falls in the range 1.75:1 to 2.25:1
        i. select the two region in the list of possible license plate regions
3. end for
4. end for

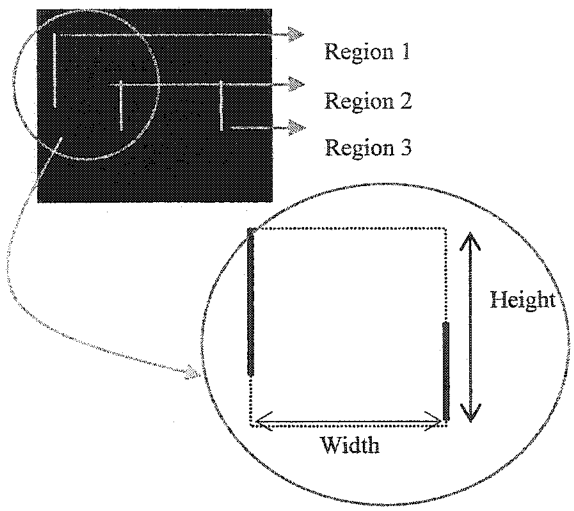

Figure 4.15 Computation of Horizontal and Vertical distances between two extracted regions.

Figure 4.15 explains how the vertical edge-matching algorithm computes the horizontal and vertical distances between two regions. In the Figure 4.15, Region 1 and Region 2 are processed for possibility that they form two vertical edges of a license plate. The width-to-height ratio of Region 1 and Region 2 certainly does not match with the specified ratio (i.e. between 1.75:1 to 2.25:1), so this pair cannot be the region of interest. In the Figure 4.14, Region 2 and Region 3 are the possible pair of vertical edges of the license plate.

In some cases, there is a possibility of more than one pair of region satisfying the above threshold. This particular case is shown in Figure 4.16 ,where there are two blocks satisfying the constraint of width-to-height ratio. The first block is certainly not the license plate. So, to overcome this problem, Black-to-White ratio of the obtained blocks is taken to get the probable license plate.

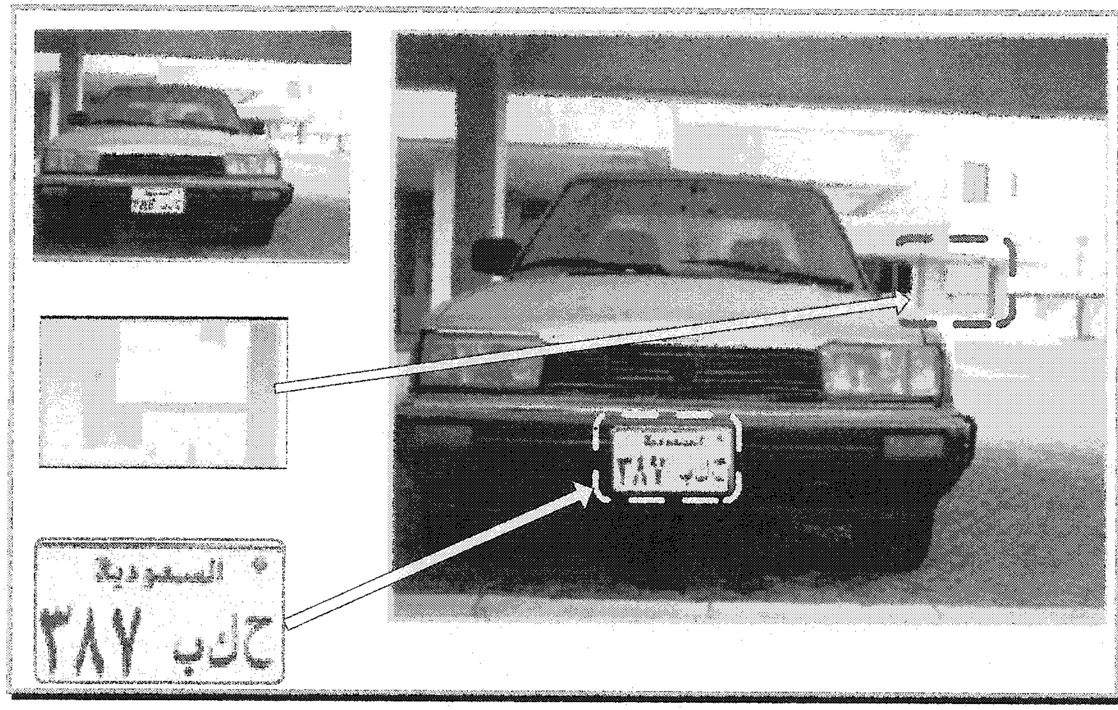

Figure 4.16 Extracted License Plate Regions of a vehicle satisfying the width-to-height constraint.

4.2.4 Black-to-White Ratio and Plate Extraction

This phase is considered when more than one pair of probable license plate regions are obtained after the matching of vertical regions for their width-to-height ratio, as shown in Figure 4.16. All the coordinate points for every pair of matched regions are stored and the black to white ratio of the stored regions are calculated with respect to the vertical edge image in Figure 4.3. Since the characters on the license plate in the Figure 4.3 contain white pixels, so the B/W (Black-to-White) ratio for the probable plate region is very less than the ratio of any of the extracted regions which does not contain a license plate. Therefore, if the pair is a possible license plate, then the ratio is within a specified threshold. The threshold is selected based on tests performed on a number of plates. For the proposed system, the threshold for B/W ratio lies between 5.5 and 11.2.

4.3 License Plate Segmentation

The License Plate Segmentation or Character Isolation is used to divide the extracted plate into individual characters [12]. To ease the process of identifying the characters, it is preferable to divide the extracted plate into six images. This is done because the Saudi Arabian license plates consist of 6 characters, with 3 letters and 3 numerals. Before the license plate is given as an input to the segmentation stage, the upper part of the plate is cropped to remove السعودية as shown in Figure 4.6. This eases the process of segmentation based on the proposed techniques. Presence of bolts and Logos in the license plates may prove to be cumbersome during the segmentation process, but the Saudi Arabian license plate template doesn't contain any logos. The bolts are usually on the upper part of the plate, and are removed in the process of cropping . For the segmentation phase the proposed strategy is based on pixel count and which is further improved by using horizontal and vertical projections. Both these methods will be discussed.

4.3.1 Pixel Count

As preprocessing to segment the characters on the plate, the plate image is first binarized with certain threshold, as is shown in Figure 4.7. A static threshold is used to achieve the required binarization. In the pixel count strategy, vertical projection profile of the cropped binary license plate is found. This is done by counting the number of black pixels corresponding to all the rows in each column. A histogram for the obtained data is plotted as shown in Figure 4.8, and the characters are segmented depending on the transition from a crest to its corresponding trough. This strategy fails in the case where there are unwanted pixel lines on the top or bottom of the license plate. The plate in Figure 4.17 depicts a thick horizontal black line at the bottom of the cropped plate. This is due to the starches and dust on the license plate. The above problem could be overcome by the isolating the plate such that there are no superfluous pixels at the top and bottom of the plate. This can be achieved by finding the horizontal and vertical projection profile of the cropped binary license plate.

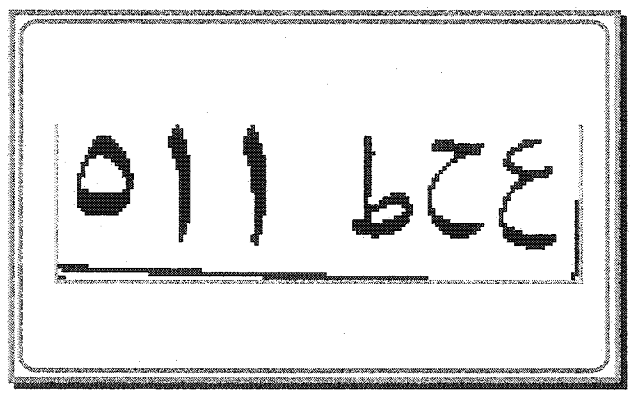

Figure 4.17 Case where Pixel Count Strategy Fails

4.3.2 Horizontal & Vertical projection Profile

The problem encountered in the pixel count strategy, is overcome by using the horizontal and vertical projections on the license plate. This method first finds the horizontal profile of the binarized license plate and the resultant plate is extracted with any unwanted pixels removed from the top and bottom of the plate. vertical projections are performed on the resultant image obtained after horizontal projection, to segment the plate into six individual characters.

Horizontal projection

The horizontal projection profile of the cropped binary plate shown in Figure 4.7 is found by counting the number of black pixels for all the columns corresponding to a particular row. The obtained count is then plotted to obtain the resultant projection for that particular row. This process is repeated for all the rows and the resultant plot gives the overall projection profile as shown in Figure 4.9. Now removing the license plate frame without any superfluous pixels is based on the assumption, that the horizontal projection has exactly one wide peak created by the rows of the characters. Therefore the start of the widest peak on the horizontal projection profile, must be the top of the characters, and the end of the peak is the bottom of the characters. The above result gives the portion of the plate containing no extra pixels at the top and bottom of the plate. This is shown in Figure 4.10. Therefore the horizontal projection profile removes any unwanted black pixels obtained due to dust or scratches on the plate. Similarly for the case in which pixel count strategy failed, as shown in Figure 4.17. The horizontal projection profile and the resultant extracted plate are shown in Figures 4.18(a) and 4.18(b) respectively.

Vertical Projection

The result obtained after horizontal projection, is given as an input to find the vertical projection profile. To obtain the vertical projection profile, the number of black pixels for all the rows corresponding to each column is found. The obtained count is plotted to get the vertical projection profile for particular column. This process is repeated for all the columns and the resultant plot gives the over all projection profile as given in Figure 4.11. In the case of Saudi Arabian license plates, there will be six projections for each of the six characters. The characters are now segmented depending on the transition from a group of black pixels to a white pixel. Figure 4.12 shows the resultant six characters. During segmentation into individual characters some threshold is taken to avoid unnecessary black pixels. Similarly in the case of Figure 4.18(b), the vertical projection profile is shown in Figure 4.18(c).

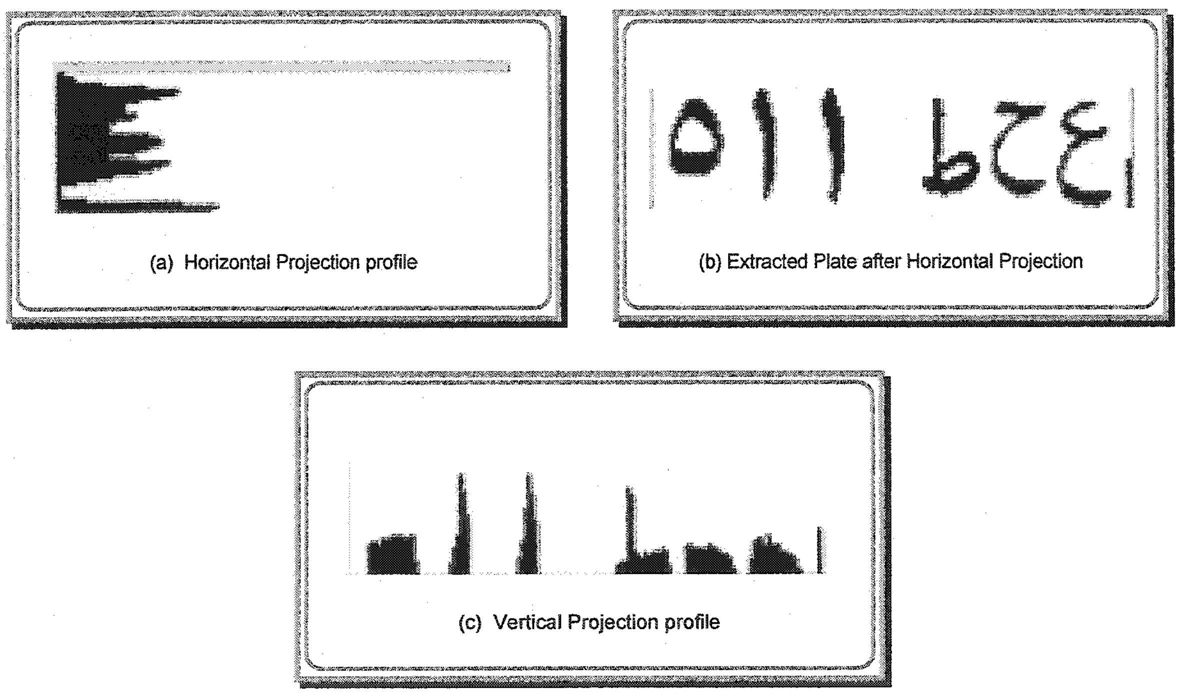

Figure 4.18 (a) Horizontal Projection Profile of image in Figure 4.17, (b) The Resultant Extracted Plate after the horizontal projection., (c) Vertical Projection Profile for the image in Figure 4.18(b).

4.4 License Plate Recognition

This is the last phase in any LPR system. This phase deals with the recognition of the segmented characters. Since the characters on the license plate contains only one font and every character is of same size, the recognition process seems to be an easy process. Generally character recognition fall into either the offline or the online category, each having its own hardware and recognition algorithms [1]. In online character recognition system, the computer recognizes the symbols as they are drawn. But offline recognition is performed after the writing or printing is complete. Our work deals with offline recognition of the segmented characters.

The process of recognizing a pattern is basically dealt with three approaches: Statistical, Syntactic and Neural approaches [39]. Statistical approach is based on decision making (i.e., the probabilistic model), Syntactic deals with the structural description of pattern (i.e., formal grammar) and Neural approach is based on training the system with a large data set of input and storing the weights (Stable State), which is used later for recognition of trained patterns. Character recognition is one of the applications in the field of pattern recognition and it generally uses syntactic and neural approaches.

4.4.1 Syntactic Approach

Syntactic pattern recognition is based on the structural information provided to the system. syntactic pattern recognition has been applied to many practical pattern recognition problems, such as, character recognition, speech recognition, fingerprint recognition, remote sensing data analysis, biomedical data analysis etc., In the pattern recognition problems, besides the statistical approach, the structural information that describes the pattern is important. This information can be used to recognize the pattern. A pattern can be decomposed into simpler sub patterns, and each simpler sub pattern can be decomposed again into even simpler sub patterns, and so on. The simplest sub patterns are called primitives (symbols, terminals) [17]. A pattern can be described as a representation, i.e., a string of primitives, a tree, a graph, an array, a matrix or an attributed string. For the proposed system, the pattern of the character is stored as a matrix. Each cell stores the information related to each pixel of a particular character. The pixel value can be either 0 (indicating a black pixel), or 1 (indicating a white pixel). The stored information is processed through the following two stages.

- Normalization of Individual characters.
- Recognition using Template Matching.

Normalization

In this phase first the extracted characters are refined to fit the characters into a window without any white spaces on all the four sides. Figure 4.19(a) shows the template of for the extracted character 'ഉ' . Now each character image is normalized to a size of 40x40. Figure 4.19(b) shows the character 'ഉ' normalized to 40x40. The normalization is done using window to view port transformation. This mapping is used to map every pixel of the original image to the corresponding pixel in the normalized image. The image obtained is variant to scaling because the characters were refined (i.e., all the white spaces on the top, bottom, left and right were removed, so as to fit the character exactly into the window).

Figure 4.19 (a) Original Template for Character,‘ ع’ (b)Normalized 40x40 Template for character ‘ع’

Template Matching Using hamming Distance

Template matching for character recognition is straightforward and is reliable. This method is more tolerant to noise. In this approach, the templates are normalized to 40x40 pixels and stored in the database. The extracted character, after normalization is matched with all the characters in the database using hamming distance approach. This approach is shown in Equation. 1.

$$\sum_{i=1}^{nrows} \sum_{j=1}^{ncols} mismatch_{i,j} \tag{1}$$

where

$$mismatch_{i,j} = \begin{cases} 1, \text{ if } original_{i,j} \neq extracted_{i,j} \\ 0, \text{ if } original_{i,j} = extracted_{i,j} \end{cases}$$

where *nrows* and *ncols* are the number of rows and columns in the original and extracted images. In our case *nrows* = *ncols* = 40, as the image is normalized to 40x40. The mismatches for each of the extracted character are found by comparing with the original characters in the database, and the character with least mismatch value is taken as the recognized character.

In the case of Saudi Arabian license plates, containing 3 alphabets and 3 numerals, recognition is done by matching each extracted character by the 27 (17 letters and 10 numerals) characters in the database. The database contains 17 Arabic letters, excluding the characters shown in Figure 4.20, because these are excluded from the license plates, and 10 numerals, Therefore the extracted normalized character must be compared with only 27 characters.

ف غ ظ ض ش ز ذ خ ج ث ت

Figure 4.20 Excluded Characters

4.4.2 Neural Approach

In this approach stable state information is used to recognize the character under investigation. In this thesis, multi layer perceptron (MLP),which is a feed forward neural network is being used. It consists of a number of layers, an input layer , several hidden layers, and an output layer. Before using MLP, features of all the characters are extracted using a feature extraction technique. The obtained features are fed to the neural network as input. In the proposed system an MLP is designed with three layers. The input layer, a hidden layer and an output layer. The first task is to extract the features of the characters and classify them with a neural network. The following subsections discuss the feature extraction method and the process of classification.

Feature Extraction

Feature extraction as defined in [10], as the problem of "extracting from the raw data the information which is most relevant for classification purposes, in the sense of minimizing the within-class pattern variability while enhancing the between-class pattern variability". Selection of feature extraction method is probably the single most important factor in achieving high recognition performance in character recognition systems. The main advantage of feature extraction is that, it removes redundancy from the data and represents the character image by a set of numerical features. These features are used by the classifier to classify the data. Different feature extraction methods are designed for different representations of characters, such as solid binary characters, character contours, skeletons (thinned characters), or gray level sub images of each individual character [19]. This section deals with the extraction of features that are appropriate to recognize individual Arabic characters. The efficiency of a feature extraction method depends on their experimental analysis. In this process, one might find that a specific feature extraction method needs to be further developed [41]. Therefore, in order to get exact performance of each of the feature extraction methods, we will have to implement all the feature extraction methods, which is an enormous task.

Moment Invariant

In this section moments and other shape descriptors by [16] have been utilized to build the feature space. Using nonlinear combinations of geometric moments Hu [16] derived a set of invariant moments which has the desirable property of being invariant under image translation, scaling and rotation.

The central moments which are invariant under any translation are defined as:

$$M_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\bar{x})^{p}(y-\bar{y})^{q} f(x,y)\,dx\,dy \qquad (1)$$

where

$$\left.\begin{array}{l} \bar{x} = \dfrac{M_{10}}{M_{00}}, \quad \bar{y} = \dfrac{M_{01}}{M_{00}} \quad \textit{and} \\ M_{pq} = \int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} x^p y^q f(x,y)\, dx\, dy \end{array}\right\} \qquad (2)$$

However for digital images the continuous image intensity function f(x,y) is replaced by a matrix where x and y are the discrete locations of the image pixels. The integrals in equation 1 and 2 are approximated by the summations

$$M_{pq} = \sum_{x=0}^{m} \sum_{y=0}^{n} (x-\bar{x})^p (y-\bar{y})^q f(x,y)\, dx\, dy$$

$$M_{pq} = \sum_{x=0}^{m} \sum_{y=0}^{n} x^p y^q f(x,y)\, dx\, dy$$

Where m and n are dimensions of image. The set of moment invariants that have been proposed by Hu [16] are given by

$$\phi_1 = M_{20} + M_{02}$$

$$\phi_2 = ((M_{20} - M_{02})^2 + 4M_{11}{}^2)^{1/2}$$

$$\phi_3 = ((M_{30} - 3M_{12})^2 + (3M_{21} - M_{03})^2)^{1/2}$$

$$\phi_4 = ((M_{30} - M_{12})^2 + (M_{21} + M_{03})^2)^{1/2}$$

$$\phi_5 = ((M_{30} - 3M_{12})(M_{30} + M_{12})[(M_{30} + M_{12})^2 \ -3(M_{21} + M_{03})^2] + (3M_{12} - M_{03})(M_{21} + M_{03}) * \ [3(M_{30} + M_{12})^2 - (M_{21} + M_{03})^2])^{1/4}$$

$$\phi_6 = ((M_{20} - M_{02})[(M_{30} + M_{12})^2 - (M_{21} + M_{03})^2] + 4M_{11}(M_{30} + M_{12})(M_{21} + M_{03}))^{1/3}$$

$$\phi_7 = ((3M_{21} - M_{03})(M_{30} + M_{12})[(M_{30} + M_{12})^2 - 3(M_{21} + M_{03})^2] + 3(M_{21} - M_{03})(M_{21} + M_{03}) * [3(M_{30} + M_{12})^2 - (M_{21} + M_{03})^2])^{1/4}$$

These functions can be normalized to make them invariant under a scale change by using the normalized central moments instead of the central moments. The normalized central moments are defined by

$$m_{pq} = \frac{M_{pq}}{M_{00}^{a}} \; where \; a = \frac{(p+q)}{2} + 1$$

These, when substituted in the above equations will give seven moments which are invariant to translation, scaling and rotation.

For each character the above moment invariant descriptors are calculated and fed to the artificial Neural Network. Table 4.3 shows the values of Φ obtained for some of the characters in the training set. Each of the character is normalized to 40 x 40 binary image.

Table 4.3: Seven Moment Invariants for different characters

	ع	ى	ق	ط	ا
Φ_1	-0.0459×10^{-6}	0.0020×10^{-6}	-0.0096×10^{-6}	-0.0146×10^{-4}	-0.0459×10^{-6}
Φ_2	0.7943×10^{-6}	0.2144×10^{-6}	0.5082×10^{-6}	0.4360×10^{-4}	-0.0519×10^{-6}
Φ_3	0.0774×10^{-6}	0.0140×10^{-6}	0.0420×10^{-6}	0.0824×10^{-4}	0.5057×10^{-6}
Φ_4	0.0792×10^{-6}	0.0141×10^{-6}	0.0414×10^{-6}	0.0794×10^{-4}	-0.0145×10^{-6}
Φ_5	0.0701×10^{-6}	0.0142×10^{-6}	0.0381×10^{-6}	0.0571×10^{-4}	0.0565×10^{-6}
Φ_6	0.1533×10^{-6}	0.0332×10^{-6}	0.0867×10^{-6}	0.1093×10^{-4}	0.0230×10^{-6}
Φ_7	0.0159×10^{-6}	0.0032×10^{-6}	0.0185×10^{-6}	0.0461×10^{-4}	-0.0540×10^{-6}

Horizontal projections

This section discusses the extraction of features based on the horizontal projection profile of individual character image. The character is represented in the form of a two dimensional matrix. The number of black pixels in each row for the character is calculated and stored in the form of a vector. The vector size is taken as the maximum number of rows among in a set of characters. In addition to horizontal projection profile, the black-to-white ratio and the height to width ratio is also taken. So the feature set given to the input layer consist of the values in horizontal projection vector plus the black-to-white ratio and the height-to-width ratio of a character. Therefore, the number of input features is equal to size of the horizontal projection vector plus two. The features are first normalized to fall within a range of -1 to +1, because the nonlinear function used in the neural network is *tansig* and it has the property that all its values lie between -1 and +1.

Classification and Recognition

Characters are classified according to their computed moment invariants by means of artificial neural networks. Among the many applications that have been proposed for neural networks, character recognition has been one of the most successful.

Many neural network architectures have been used in OCR implementation. MLP is usually a common choice. MLP have been applied successfully to solve some difficult problems by training them in a supervised manner with error back propagation algorithm [14]. This algorithm is based on error correction learning rule. Basically, error back propagation learning consists of two passes through the different layers of the network: a forward pass and the backward pass [14]. In the forward pass an input vector is applied to the nodes in the input layer, and its effect propagates through the network layer by layer. Finally a set of output is produced as the actual response of the network. During the forward pass the weights of the network are all fixed. During the backward pass, the weights are all adjusted in accordance with the error correction rule, i.e., the actual response of the network is subtracted from the desired response to produce an error signal. This error is then back propagated through the network and the weights are adjusted to make the actual response of the network move closer to the desired response.

In the proposed system, the MLP Network is implemented with three layers. An input layer , a hidden layer and an output layer of linear neurons. The number of neurons in the input layer varies depending on the size of feature vector to be used. In the case of moments invariants proposed by Hu [16], seven features are used as input. For the case when horizontal projections are used, the feature set depends on the maximum number of rows of a character in the given character set. The hidden layer consists of ten neurons. The number of the output neurons depends on the number of the characters in the character set. Since the system consist of 27 characters (17 Arabic alphabets and 10 numerals), the output for each character can be represented using 5 bits. Therefore, the output layer consists of five neurons, each representing a bit as an output. For example, character 'ﭗ' is represented by 0 0 0 1 0. For the proposed system, a three layer MLP, as shown in Figure 4.21 is used for the classification of characters.

The recognition rate was evaluated for the features obtained by applying the Hu's moments [16], and the horizontal projections. The recognition rate of about 75% was achieved by using the seven moment invariants proposed by Hu. This low recognition rate is due to the fact that the features obtained for each of the characters are very close. Therefore it was very difficult for the neural network to classify the characters correctly. The results were further improved by using the later approach (i.e., horizontal projection profile of each character). It gave a recognition rate of about 85%. The results for all the 27 characters using both the techniques were plotted. First the neural net was trained with 70% of the available data and the remaining data was used for testing. The tested results were plotted for both the feature sets. In the case of Hu's moments, the resultant plot for recognition is shown in Figure 4.22. On the other hand when horizontal projection is used, the resultant recognition rate is shown in Figure 4.23.

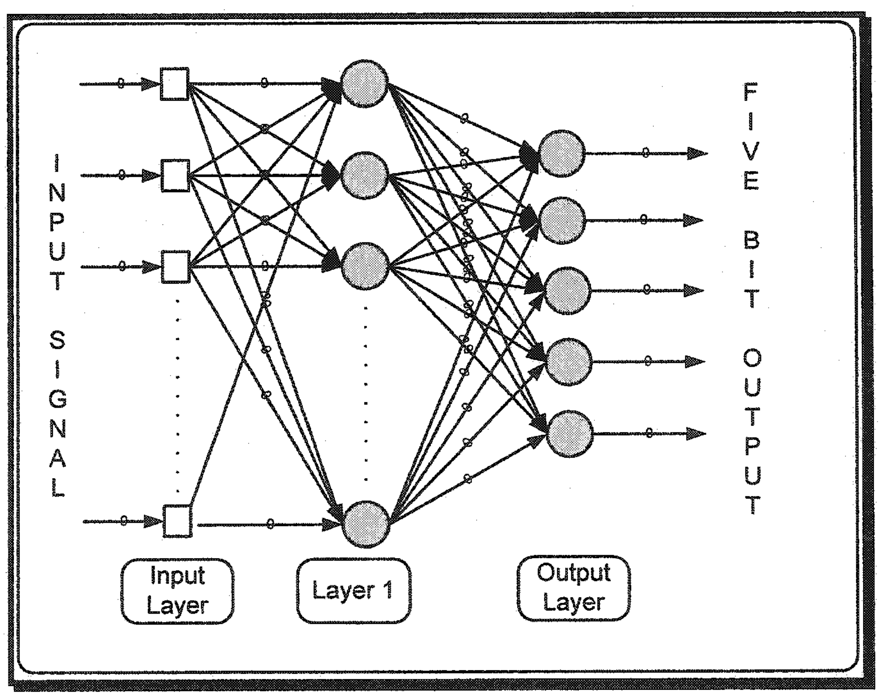

Figure 4.21 Three layer MLP

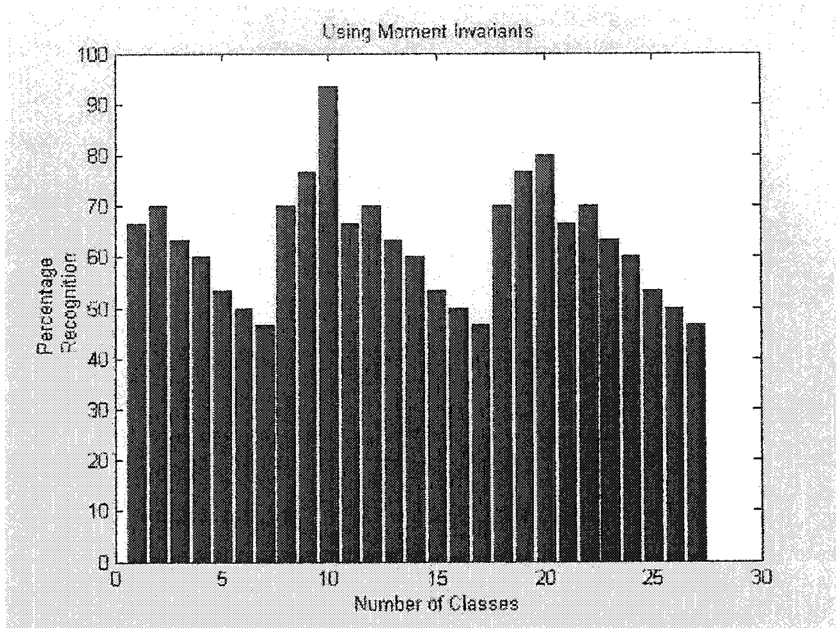

Figure 4.22 Recognition rate for each class of character using seven moment as the features.

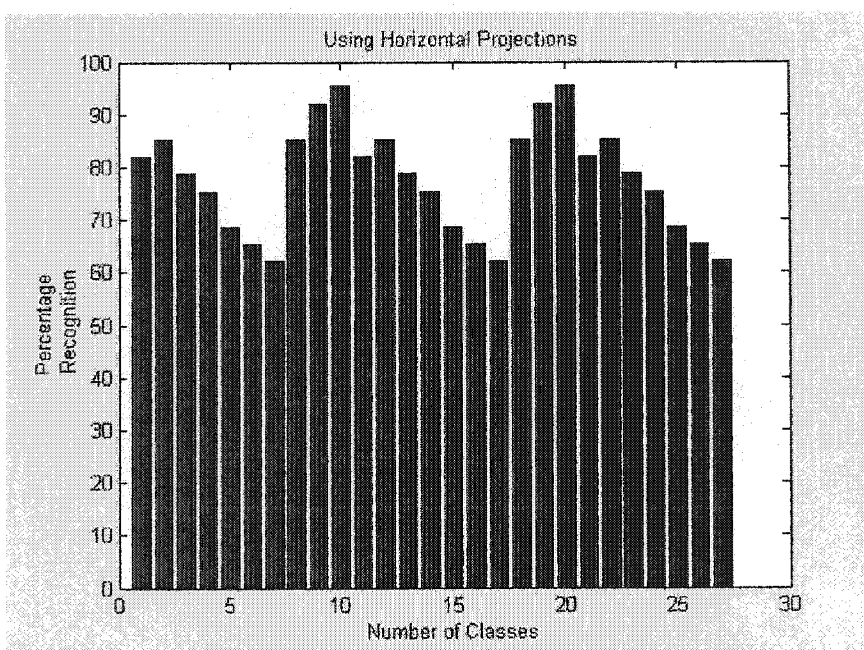

Figure 4.23 Recognition rate for each class of character using horizontal projections as the features.

4.5 Summary

This chapter discussed the conceptual model of the proposed LPR system. The implementation issues related four stages: Image acquisition, License plate extraction, segmentation and recognition were discussed in detail. A high resolution digital camera was used to acquire good quality image. The gray scaled version of the input image was passed to the extraction phase, which consists of four sub phases : vertical edge detection , filtering, vertical edge matching and calculating the black-to-white ratio. The extracted plate was fed as input to the segmentation stage to get the segmented characters. Segmentation was implemented using pixel count strategy, which was later improved by finding the horizontal and vertical projection profile to obtain better results. As the last phase in the conceptual model, recognition using both the syntactic and neural approaches was presented. Lastly, the recognition rate is plotted for both Hu's moments and horizontal projection.

CHAPTER 5

EXPERIMENTAL ANALYSIS AND RESULTS

5.1 Introduction

This chapter describes the tests performed to verify the whole process of a license plate recognition system. The tests performed using different techniques for the extraction, segmentation and recognition phases will be discussed. A comparison of the existing techniques with the proposed ones is also presented. This chapter is divided into four sections. The first section presents the comparison between the existing and the proposed technique for the license plate extraction phase. Next section deals with different segmentation techniques. Followed by the segmentation is the recognition of individual character, which is dealt with the syntactic and neural approaches. After the test on the recognition phase, an overall system test is shown in the next section. Finally, the chapter concludes with a brief summary.

5.2 Extraction Test

In the extraction phase a test is performed and results are shown by comparing the proposed method for extraction with the method proposed by [22] using Hough Transform. Hough transform is a technique which can be used to isolate features of a particular shape within an image. It is generally used to detect regular curves such as lines, curves etc. In [22], Hough transform was used to detect horizontal and vertical line segments. The extracted line segment is then used to find the license plate using the ratio of length of horizontal and vertical lines that match the license plate.

The proposed method for the extraction of the license plate is based on four steps: Vertical edge detection, Filtering, Vertical edge matching and finding the B/W ratio. Our strategy is based on finding only vertical edges and comparing each pair of edges and then finding the B/W ratio to get the resultant plate.

5.2.1 Criteria of success

When testing the license plate extraction of Hough transform and the proposed method, it is important to realize, that the goal is not to provide a single image, but rather a limited list of regions, where at least one of the regions should contain the license plate. The criteria for when a region is useful, is relatively straight-forward. It must encompass the criteria for when a region is useful, is relatively straight-forward. It must encompass the entire license plate, so that all six characters are fully visible. Also, it may not contain too much noise in the form of edges.

In the case of Hough transform, If there are many regions containing more than just the license plate, it will be harder to distinguish the correct region from the entire list of regions. The method should not provide an excessive amount of regions. A very large amount of regions makes it harder to select the most probable license plate candidate but in the case of the proposed strategy, size-and-shape filtering is used to get the resultant regions, which is discussed in Chapter 4. A very large amount of regions does not have any effect in mis-detecting the regions containing the license plates of regions, unless the image under consideration is of bad quality.

5.2.2 Test Data and Test Description

Both methods were tested using two sets of images. These images were all taken under the same lighting conditions, but shadows from trees nearby gave some variance in the brightness of the license plates. Also, the images were taken at various distances from the vehicles, to ensure that the algorithms were constructed to be invariant to the size of the license plate.

As mentioned, the test was performed on all images. For each image, a manual inspection was made, to see if the license plate had been successfully extracted in at least one of the regions. This rather overwhelming task was eased by first trying to automatically identify the correct license plate, so that in many of the cases, it was not necessary to go through all of the regions. For both methods, the number of extracted regions was registered.

Generally, a lower number will make the task of finding the most probable candidate easier later on. Hough transform results in more number of regions then the proposed method.

5.2.3 Results

The proposed method based on vertical edge matching turned out to be very robust way of finding the license plate. As Table 5.1 indicates, it almost always found the plate. In only 4 of the 72 test images, it was not capable of locating a satisfying region. This is because, the edges of the plate were not detected properly. This is due to the bad quality of the acquired image. A further scaling of the threshold value might have solved this issue, but this might not provide better overall results. The total amount of extracted regions are very less, this is due to the size-and-shape filtering applied on the vertical edged images. Filtering removes all the unwanted vertical edges, which does not form a part of the license plate region. The details regarding the extraction of license plates were discussed in chapter 4.

In the case of Hough transform, the percentage of extracted plates was somewhat less. This is due to the fact that the number of candidate regions was approximately a factor 2 higher than that of the proposed method. Since this method discussed in [22] finds both the horizontal and vertical edges of the plate and then compare their lengths to match it with the standard license plate, extracting the plate region. This method may fail if there is more than one region satisfying the ratio of standard plate. It is also very difficult, if the number of horizontal and vertical regions is very large. Finally, this method can only be applied to images containing license plates, without any angle of deviation.

As is also seen in Table 5.1, the amount of regions found by the Hough transform is more. The reason is, that a parking lot in the background of some of the images causes a very high number of edges to be detected.

In conclusion, the proposed method will guarantee a very high success rate, but also a rather large percentage of the regions can easily be discerned as random regions without much similarity with a license plate by applying size-and-shape filtering , before finding the probable regions. Finally the region containing the plate is extracted from the obtained probable regions.

The algorithm was proven to be almost as efficient when applied to images, under different conditions, i.e., taken in different lighting situations and distances, which is discussed in the last section 5.5 of this chapter.

Table 5.1 Results for Hough Transform and The Proposed Method

Method	**Success Rate**	**No. of Regions**
Hough Transform	46/72	50-100
Proposed Method for Extraction	68/72	20-30

5.3 Segmentation Test

This section describes the test for isolation of character using both the pixel count and horizontal and vertical projections. When testing the license plate segmentation using the pixel count and the horizontal and vertical projection profile, it is important to realize, the success rate of both these methods and the criteria for their failure.

Pixel count method simply searches for the characters by finding peaks and valleys in a vertical projection. Where as in the later case, first the horizontal projection profile is found. Since the profile obtained has exactly one wide peak created by the rows of the characters. Therefore the plate is extracted from the start of the widest peak to the end of that peak. This result gives a plate without any noisy pixels on the top and bottom of the plate. The tests on these methods are provided along with the results.

5.3.1 Criteria of success

The purpose of the method is to divide the extracted license plate into exactly six sub images, each containing one of the six characters. A successful isolation fulfills all of the following criteria:

- The plate must be divided into exactly six sub images.
- None of the six characters may be diminished in any way.
- The sequence of the sub images has to be in the correct order. This means that the first character of the plate is the first of the sub images.

An example of an unsuccessful and a successful isolation is seen on Figure 5.1.

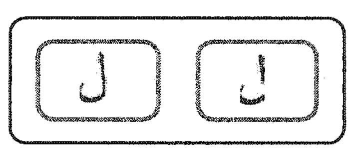

Figure 5.1: Example of unsuccessful and successful isolation.

5.3.2 Test Data and Test Description

The methods for isolating the characters in the license plate described in Chapter 4 have been tested. The test is a simple black box test. A series of input images is given to the algorithm, and the success of the test simply depends on the resulting output images. The test has been performed on both the pixel count and the method using horizontal and vertical projections. Both the methods are tested independently.

The test is performed on images received from a successful extraction process, meaning that each image does contain a readable license plate. As in the test of the extraction process described in Section 5.2 , the isolation process was tested using 72 sets of input images.

5.3.3 Results

Both the methods are tested on images with no quality improvement (no preprocessing) as well as images which have undergone a quality improvement, such as use of dynamic threshold. First the method was tested on images without using any form of quality improvement. Next the quality of the image was improved by setting certain dynamic threshold, using a Matlab function, *imadjust()*. Thereby the quality of binary images improved drastically and this reflects clearly on the result of the test. The results proved to almost identical for both the methods. The results for both the methods under different conditions are shown in Table 5.2.

Table 5.2: Results from test on pixel count method and Horizontal-and-Vertical Projections

Technique	Method	Successful	Percentage
Pixel Count	Without Preprocessing	52 of 72	72 .2%
	Dynamic Threshold	65 of 72	90.2%
Horizontal-and-Vertical Projections	Without Preprocessing	60 of 72	83.33%
	Dynamic Threshold	69 of 72	95.8%

5.4 Recognition Test

The final step in recognizing a license plate is identifying the single characters extracted from the segmentation stage. Two approaches for recognition were presented earlier in chapter 4. The first method is based on syntactic approach and the next one is based on the neural approach.

5.4.1 Criteria of success

For this test to be a success, as high a percentage as possible of the characters must be identified correctly. A few errors are acceptable, since a full implementation should be able to notify the operators, when it is unable to identify a character, allowing for manual identification. Since as many characters as possible should be correctly identified. The success criterion for the character identification test is a hit rate approaching 100 %.

5.4.2 Test Data

The test data originates from the 106 test pictures that were extracted and isolated during the previous tests. The actual test data are the single characters extracted from the license plate, as mentioned in Section 5.3. The letters are identified using template based identification and a neural approach by extracting relevant features of a character.

5.4.3 Results

Results of the tests with syntactic approach, (i.e., getting the pixel information about a character and using the template matching using minimum hamming distance to find the recognized character) and the neural approach using the two different features extractors are shown.

The test using the template matching with the minimum hamming distance was tested on a large set of input data (106 segmented character images). First the system was trained on the available data set and based upon the training data a database of different classes of characters was created. Then the system was tested on the new data set to get the overall recognition rate. In the case of the proposed system, there are 27 set of classes for the 27 characters used on the license plate. The results are shown in Table 5.3.

A result of the test using neural approach was done using two different feature extracting techniques. As the first feature extractor seven moments proposed by Hu [16] was used. The details about this are provided in Chapter 4. The result obtained using this features was very less, due to the closeness of features for almost all the characters. To improve the feature extractor method, a set of horizontal projections for a character was taken as the second feature extractor. This method showed an improvement to the previous one. In both the cases 100 different data sets for each of the 27 classes of characters was created. This data set was created by adding some noise information to the 10 existing original sets. From this set of data, 70% was used to train the neural net and 30% was used for testing. The results for the recognition on the tested data using Hu's moments and Horizontal projections is shown using the bar graph in Figures 5.2 and 5.3 respectively. The overall recognition rate using both the syntactic and neural approaches was shown in Table 5.3.

Table 5.3 Results of the test performed on 106 training set of each class of characters.

	Syntactic Approach	Neural Approach	
		Moments	Horizontal Projections
Correct Recognition	100/106	79/106	91/106
Percentage Recognition	94.33%	74.5 %	85.8%

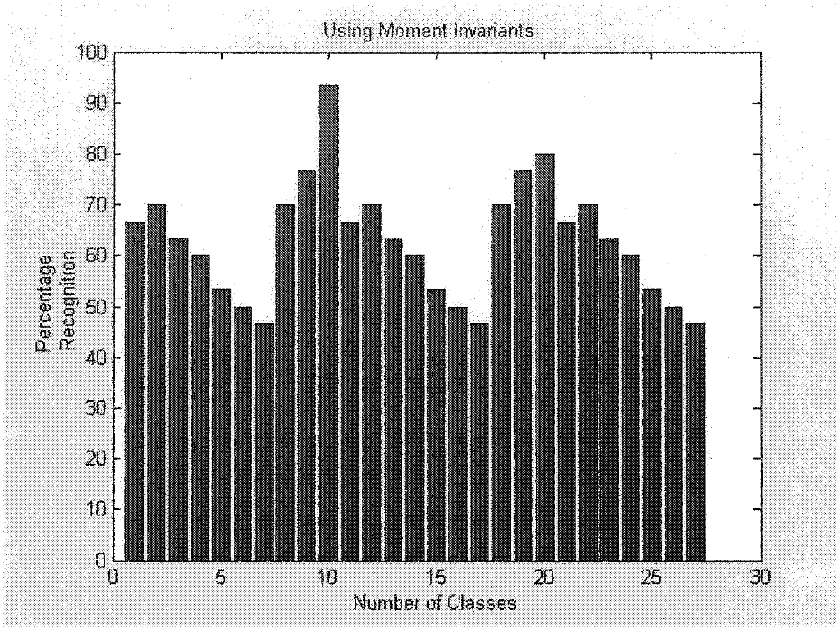

Figure 5.2 Recognition rate for each class of character using seven moment as the features.

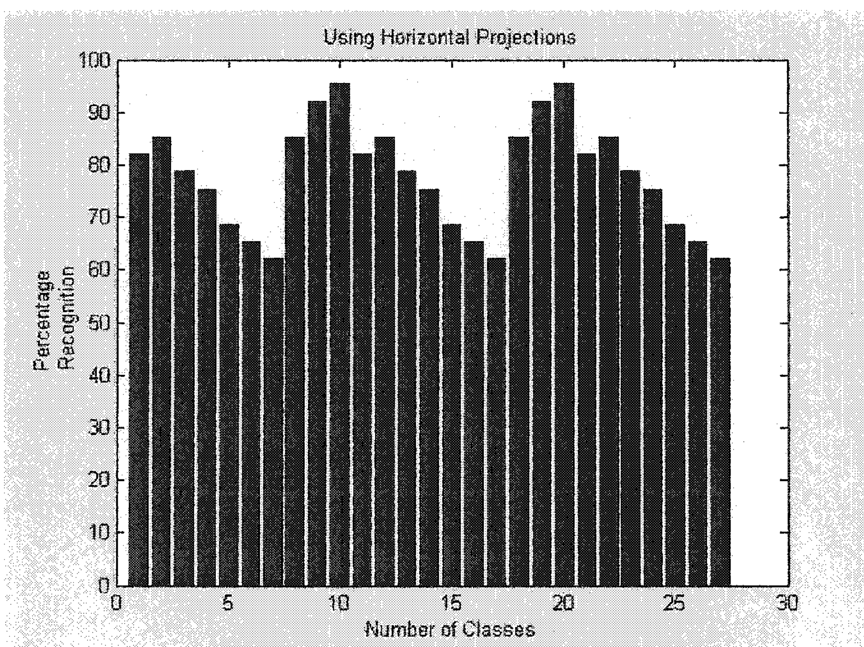

Figure 5.3 Recognition rate for each class of character using horizontal projections as the features.

5.5 System Test

In the previous sections, the components of the system were examined separately in terms of performance. It is also relevant to test the combination of the components. When testing the individual components, only useful input was used. In real life, an error made in the license plate isolation will ripple through the system, and affect the overall performance. This section will describe the tests performed on the final system in different conditions.

Experiments have been performed to test the proposed LPR system. The system is designed in MATLAB6.1 [26] for recognition of Saudi Arabian license plates. The image to the system is a gray scale of size 640x480. The test images were taken under various illumination conditions (i.e. noon and in the evening).

The experiments for the above system were performed under different situations, i.e. the experiments were carried out for the following cases.

- License plates in normal shapes. (Shown in Figure 5.4.)
- License plates that are leaned with some angle of view. (Shown in Figure 5.5.)
- License plates that have similar color to license plate body. (Shown in Figure 5.6)
- Dirty license plates. (Shown in Figure 5.7)

The whole system is working fine achieving high recognition rate. Table 5.4 gives results for the percentage recognition of license plate extraction, segmentation, and character recognition. A failure results if a character of the license plate is incorrectly interpreted. This normally happens due to bad quality of extracted plate. It is shown that the system has correctly recognized 581 license plates out of 610 test images. The system was implemented on PIII (700MHz) using MATLAB6.1 [26].

The experimental results show that the shortcoming of the proposed system is mainly due to bad quality of input image during the acquisition stage (i.e., the bad quality is due to the presence of dirty or unclear license plates), or unclear detection or extraction of the edges. The segmentation phase gives good results, because the image is converted into binary form. In character recognition there is some form of mis-recognition. This is due to the nuts and bolts within the character. Otherwise, there is a standard font for all the Arabic characters used in the license plates, which do not cause any problem during recognition phase.

Table 5.4. Recognition rate for license plate extraction, license plate segmentation and license plate recognition.

	License Plate Extraction	License Plate Segmentation		License Plate Recognition	
		Pixel Count	HPP & VPP Projection	Syntactic Approach	Neural Approach
Correct Recognition	298/310	280/310	297/310	295/310	264/310
Percentage Recognition	96.22%	90.03%	96.04%	95.24%	85.16%

5.6 Summary

This chapter covered the test on the individual phases of an LPR system with some of the comparisons with the existing ones. For the extraction phase the proposed method showed a better result then extraction by Hough Transform. In the segmentation test, the horizontal and vertical projection profile gave better result than the pixel count strategy. For the case of recognition, the template matching approach using hamming distance proved o be effective, because the recognition was based on individual characters of same font. The neural approach was unable to give better results due o the closeness of the features for all the class of characters. The neural approach can be improved by using good feature extraction techniques. Then an over all system performance was also presented achieving good results.

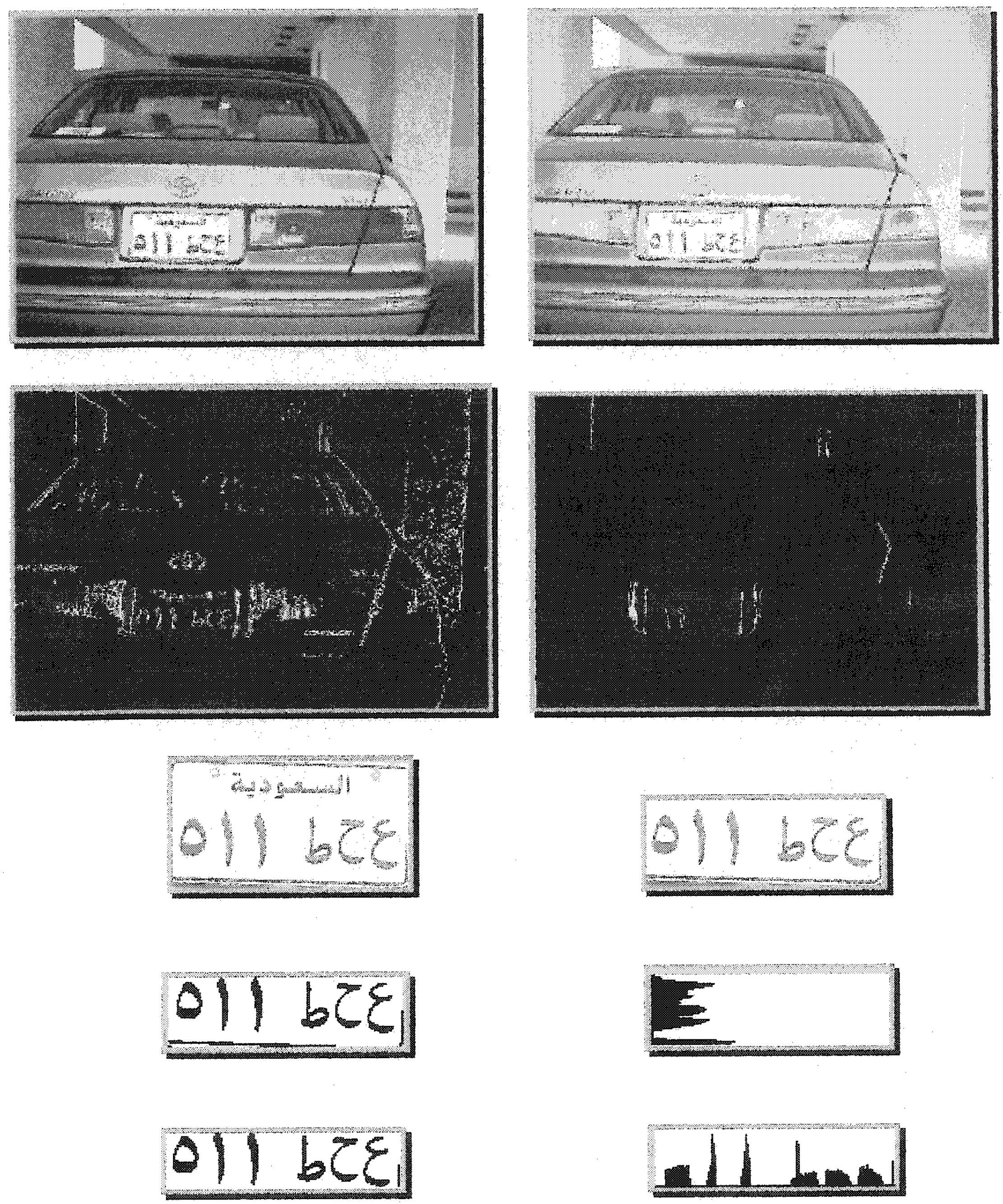

Figure 5.4. The Overall Process of an LPR System with license plate in normal form showing the rear end.

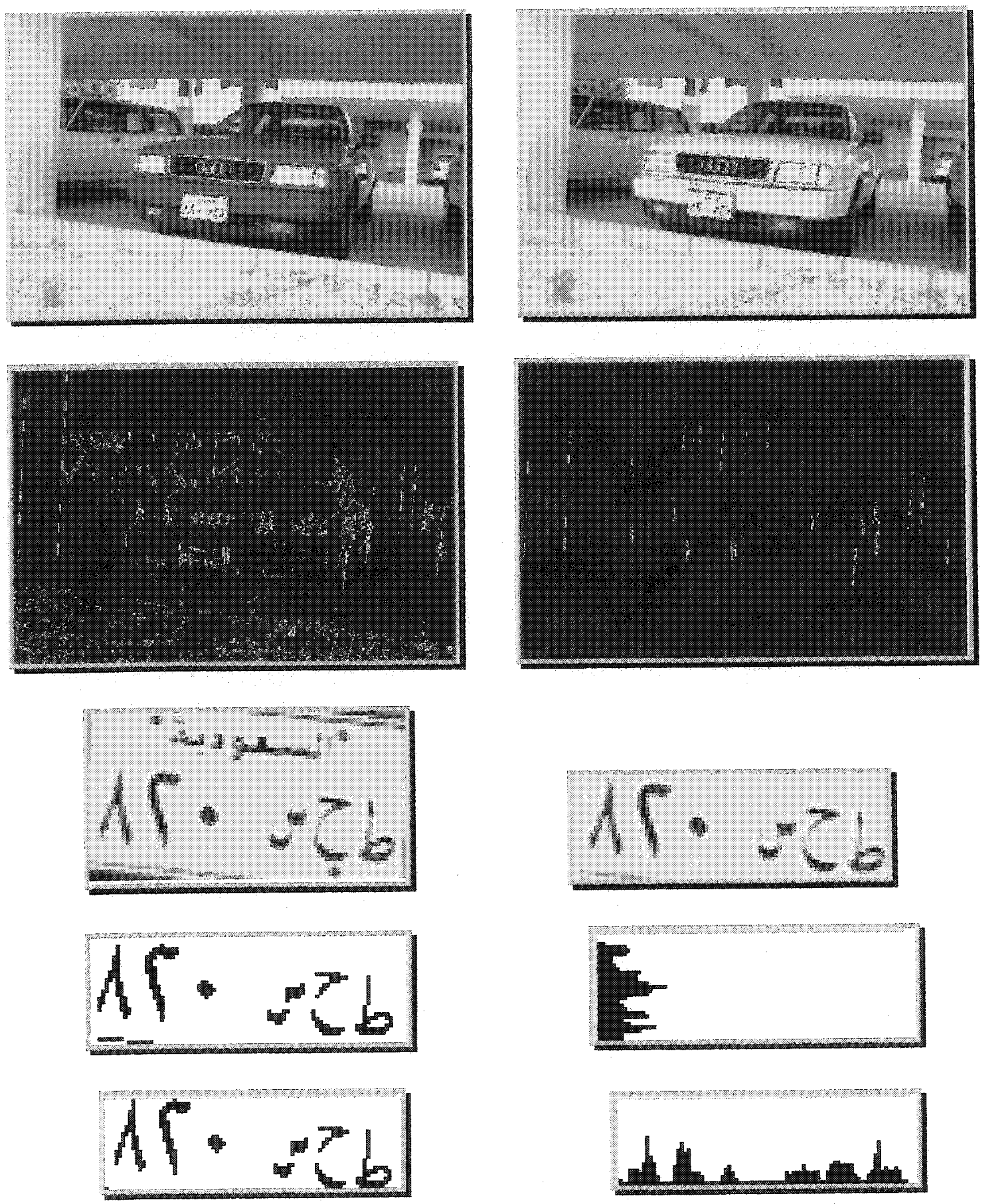

Figure 5.5. The Overall Process of an LPR System with the car slanted towards right.

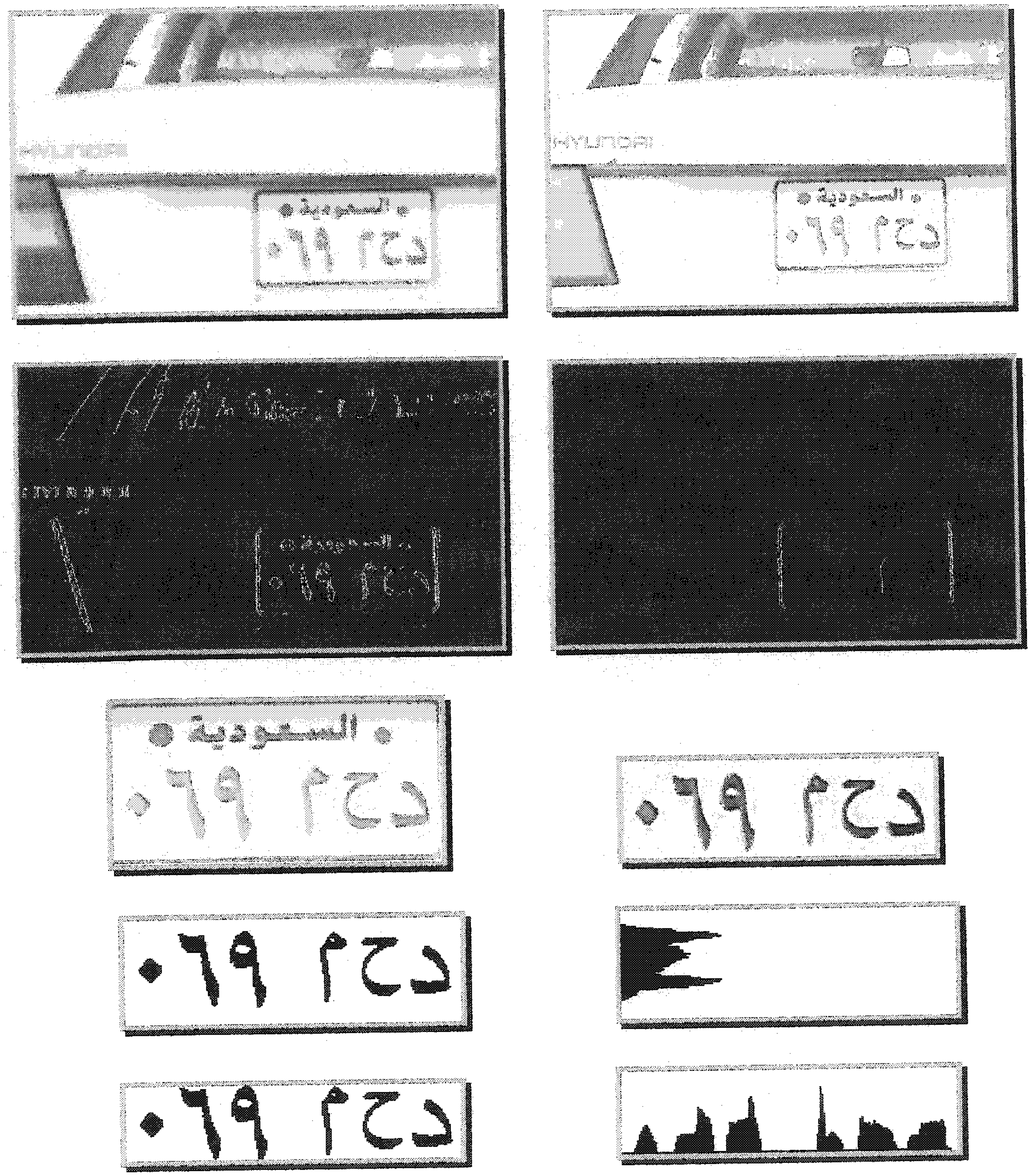

Figure 5.6. The Overall Process of an LPR System showing a car having similar color to license plate body.

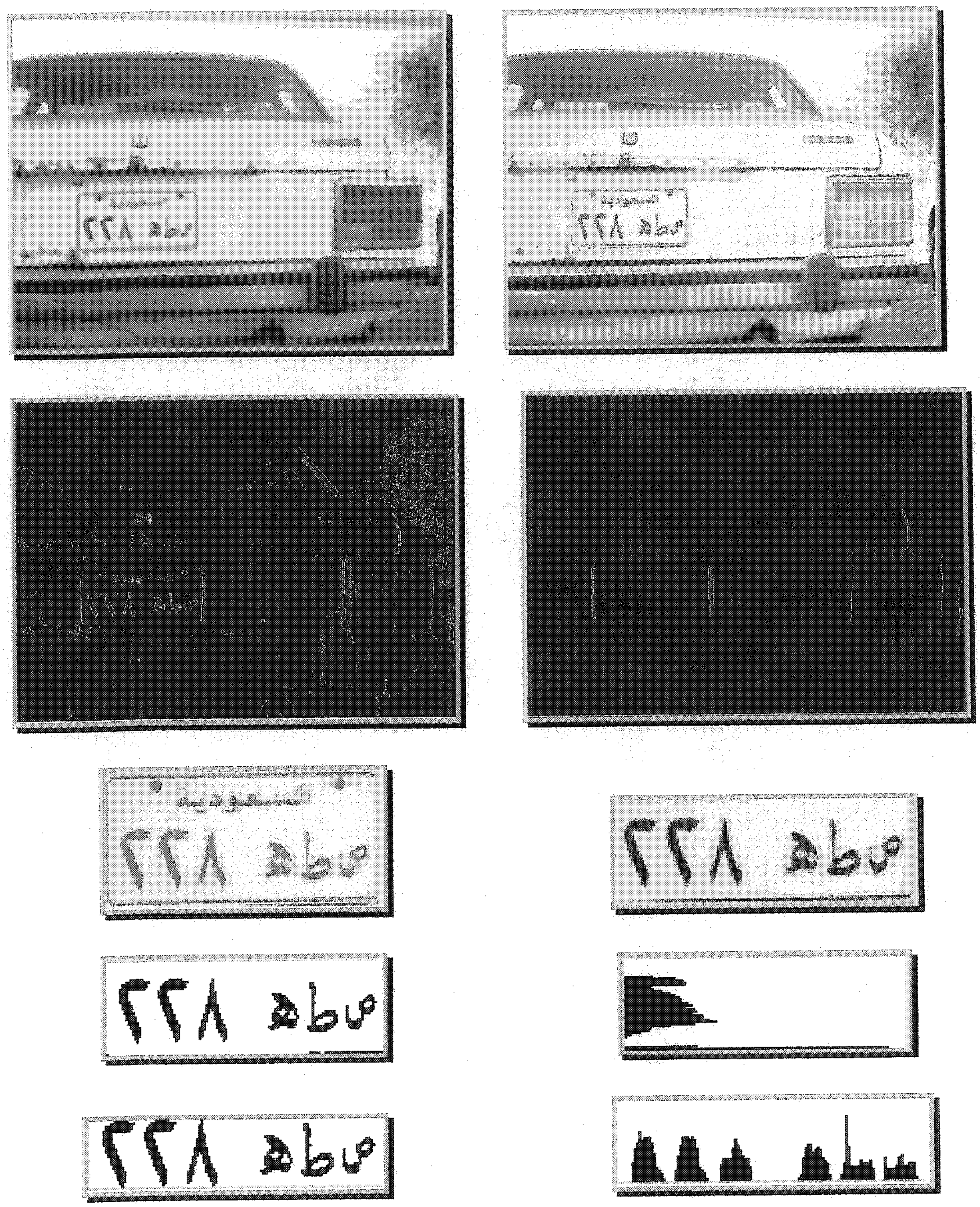

Figure 5.7. The Overall Process of an LPR System showing a car with dust and scratches.

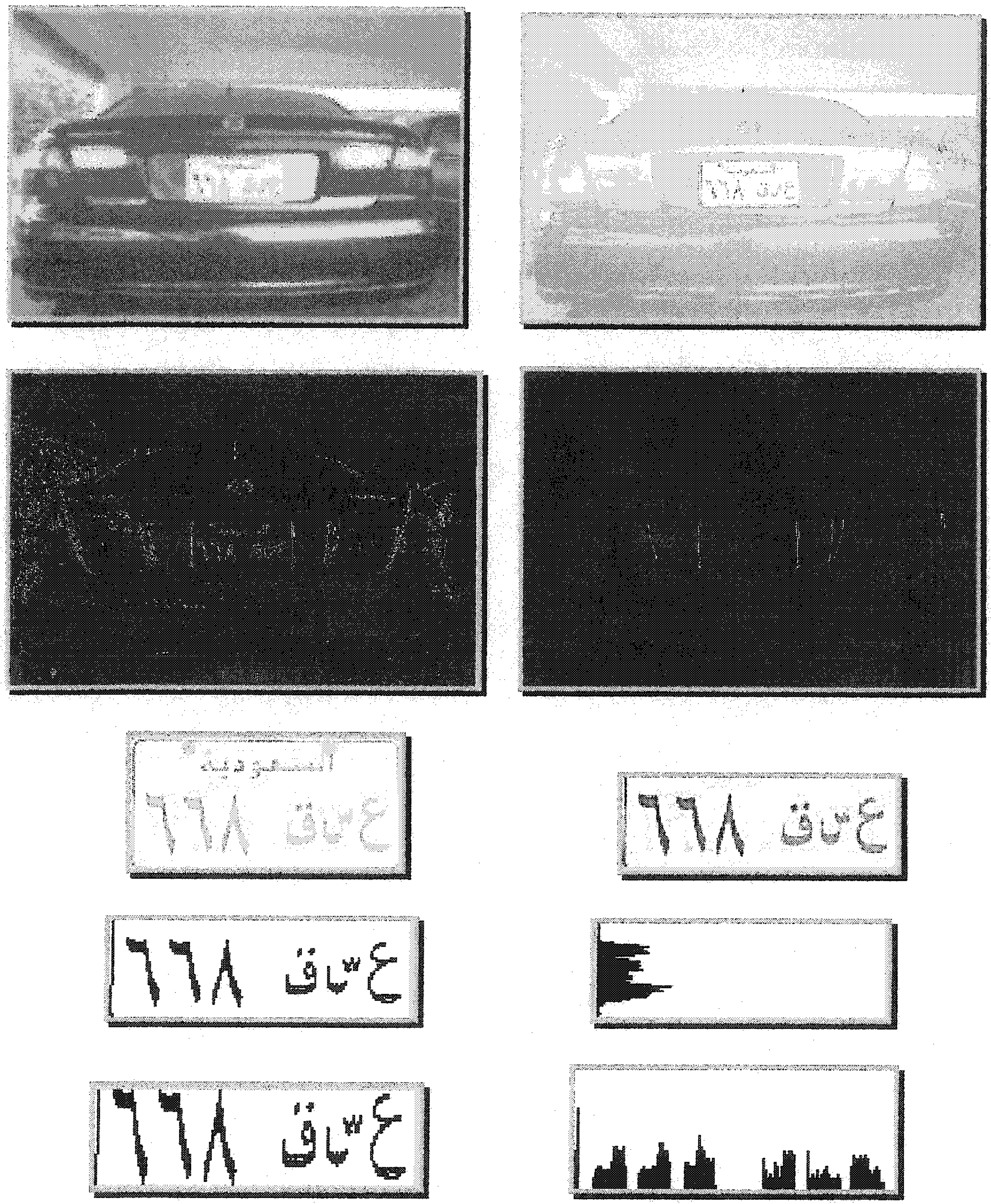

Figure 5.8. The Overall Process of an LPR System for an image taken under sunlight.

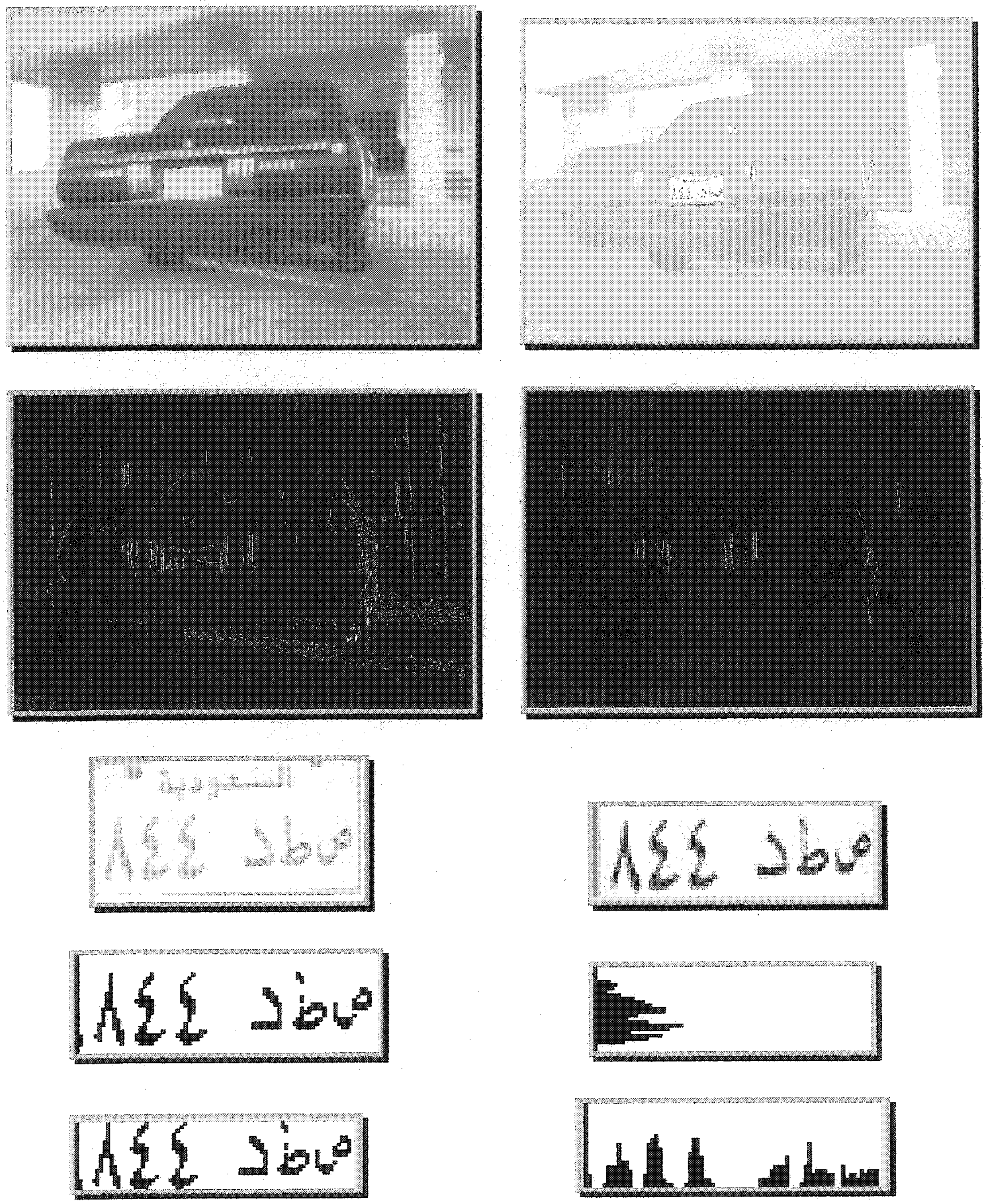

Figure 5.9. The Overall Process of an LPR System with the rear part of the car slanted towards right taken before sunset.

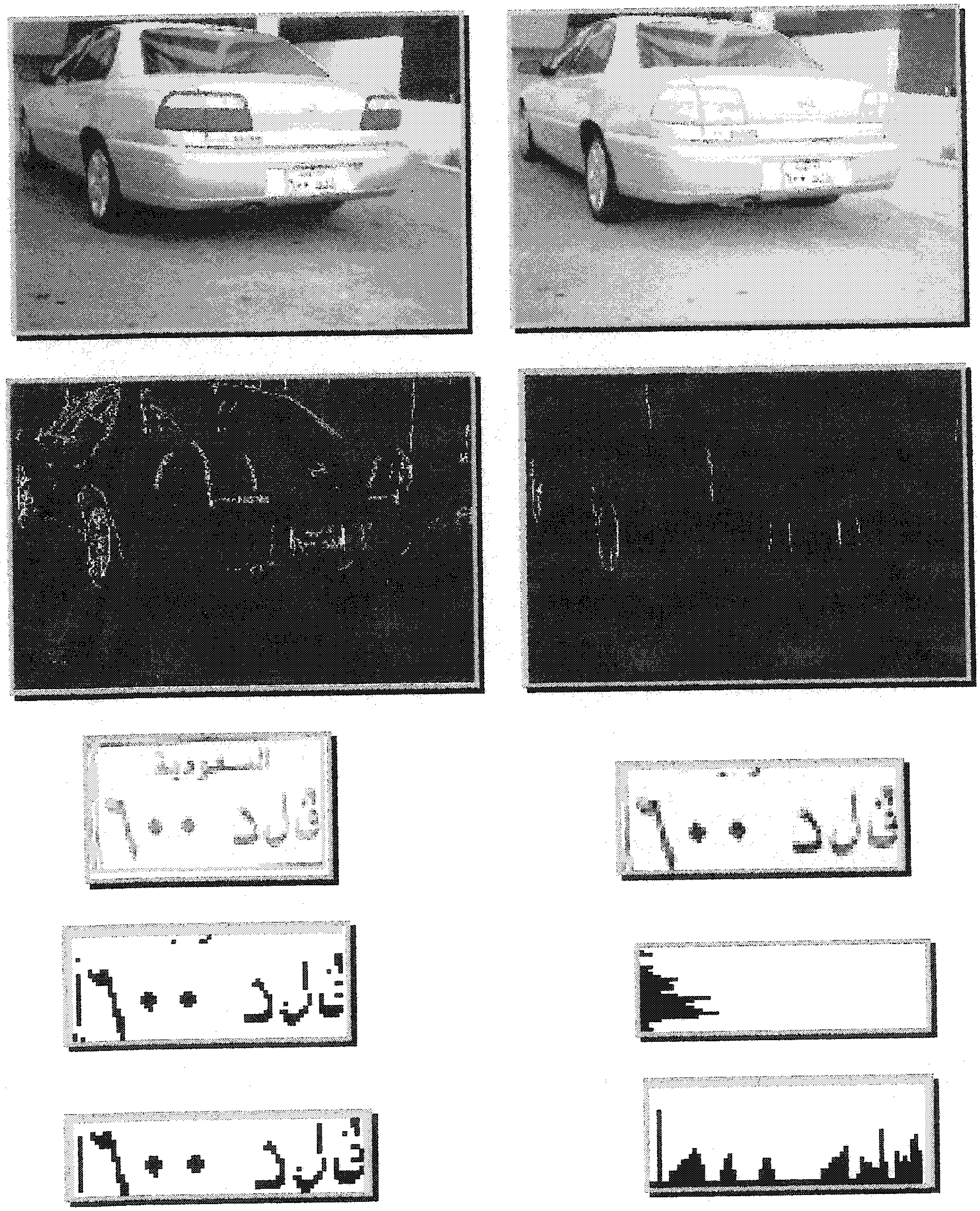

Figure 5.10. The Overall Process of an LPR System with the rear part of the car slanted towards left.

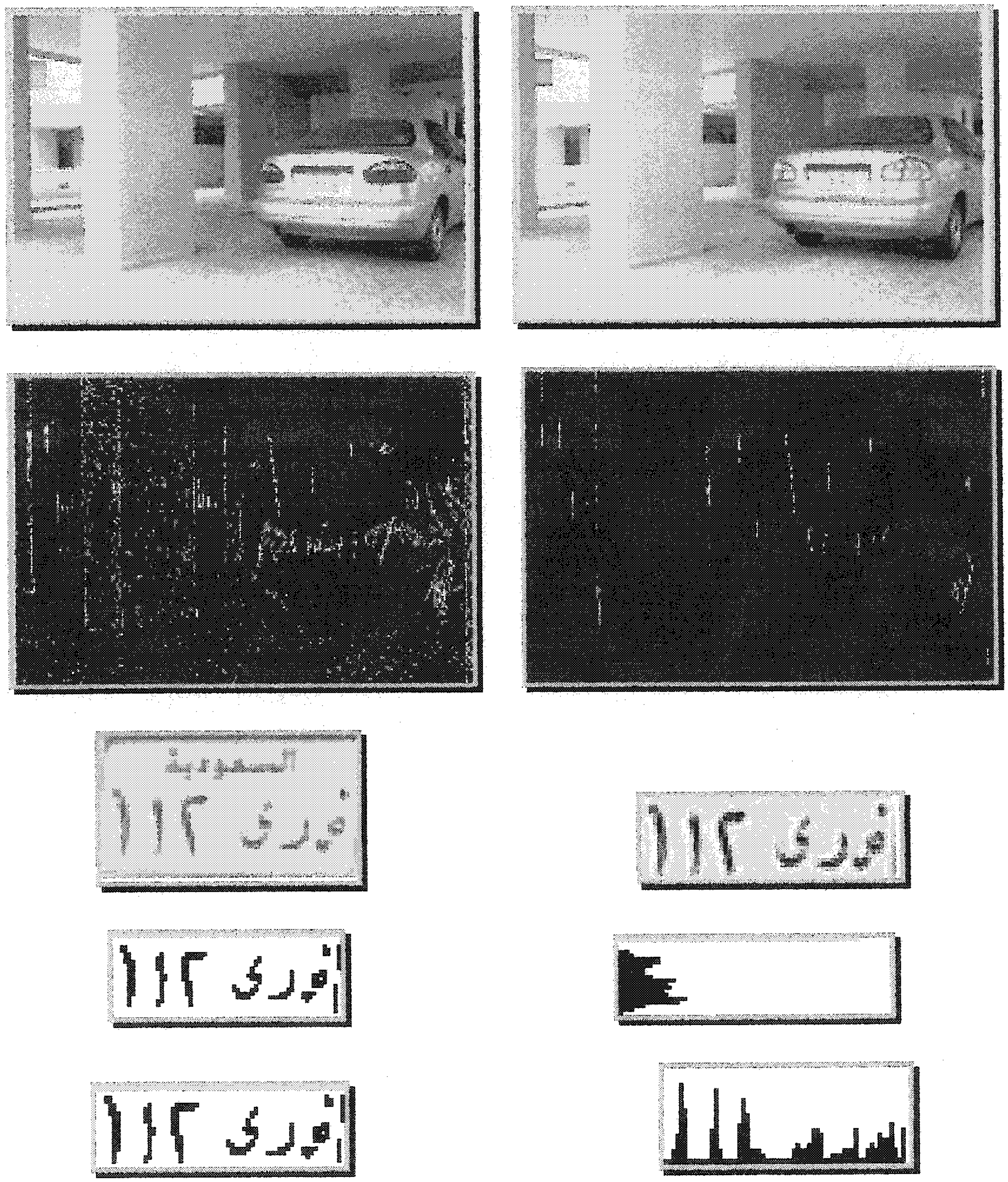

Figure 5.11. The Overall Process of an LPR System with the rear part of the car slanted towards right.

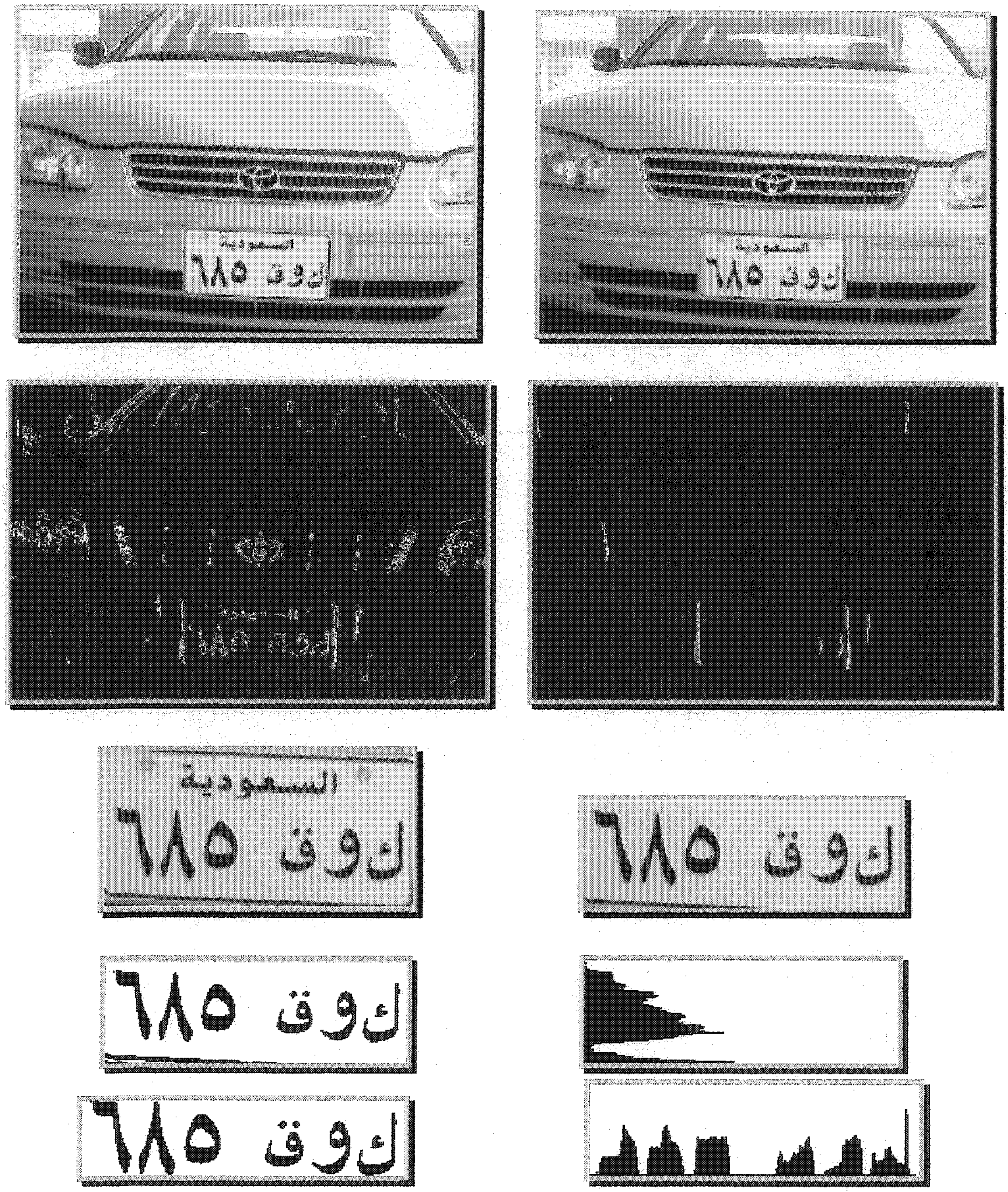

Figure 5.12. The Overall Process of an LPR System with the front part of the car with a closer look of the picture.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

The purpose of this thesis has been to investigate the possibility of making a system for automatic recognition of license plates. This thesis was aimed at contributing towards the research in the field of computer vision and pattern recognition. The current system investigates the possibility of automating the whole process of license plate recognition Given an input image, it should be able to first extract the license plate, then isolate the characters contained in the plate, and finally identify the characters in the license plate. For each task, a set of methods were developed and tested. For the extraction part, Hough transform and Vertical edge matching were discussed. But the later approach proved capable of extracting the plate from an image.

The methods developed for isolating the characters proved very reliable. The method for finding pixel count was not much successful, but the horizontal and vertical projection profile of the plate gave promising results.

For the actual identification process, two approaches were used; one based on template matching and the other on neural network. Both methods proved to give good results but the template matching using hamming distance was highly successful, giving good recognition rate. Neural network approaches were based on extracting features using two methods, First was based on extracting features by seven moment invariants proposed by Hu [16] and the other was to find the horizontal projection profile of a character as the feature. The low recognition rate in the case of neural network was due to the close features for the set of 27 characters.

In order for the system to be useful, it should be able to combine the three different tasks, and to recognize the license plates in a high percentage, so the use of manual labor is reduced as much as possible. This implies, that the success rate for the individual parts should be close to 100 %. The results obtained are summarized in Table 6.1.

Table 6.1 Main Results

Phases	**Success Rate**	**Successful**
Plate Extraction	97.1 %	✓
Character Isolation	100 %	✓
Character Identification	98.1	✓
Overall performance	87 %	✓

As can be seen, the individual parts perform very satisfactory, all with a success rate close to 100 %. The plate extraction succeeds in 97.1 % of the test images, and this is a very high success rate. The extraction fails for a few number of the images. This is acceptable, since the extraction works in more than 97 % of the images, thereby fulfilling the criteria of this task. The part of isolating the characters contained in the license plate succeeds in 100 % of the cases, and thus is very successful, achieving the goal set for this task.

Out of the isolated digits, 98.1 % were correctly identified. This is also a very high success rate, and it must be taken into account. The overall performance is not as high as for the individual tasks, but still a large amount of license plates is correctly identified, namely 87%. In general, the conclusion of this thesis is, that a system for automatic license plate recognition is constructed.

6.2 Future Work

The following points are suggested as recommendations for the future work

- The present system is designed to work for images taken from a certain distance (1 to 3 Meters), making the system distance free can be a good extension.
- The proposed system works for static car images, the image acquisition stage can be improved so as to capture images of moving vehicles.
- The recognition using neural approach can be improved using a good feature extractor algorithm or improving the present feature extraction technique.

- Finally, the adopted approach with the above suggestions can be generalized to work for all the Arabian license plates.

REFERENCES

[1] Amin, A., "Offline Arabic Character Recognition: The state of the art", *Pattern Recognition*, vol. 31, no. 5, pp. 517-530, 1998.

[2] Bailey, D. G., Irecki, D., Lim, B. K., and Yang, L., "Test bed for number plate recognition applications", *Proceedings of the First IEEE International Workshop on Electronic Design, Test and Applications (DELTA'02), IEEE Computer Society.*, 2002.

[3] Belkasim, S.O., Shridhar, M., and Ahmadi, A., "Pattern Recognition with moment invariants: A Comparative study and new results," *Pattern Recognition*, vol. 24, pp. 1117-1138, 1991.

[4] Bokser, M., "Omnidocument technologies", *Proceedings of the IEEE*, vol. 80, pp. 1066-1078, 1992.

[5] Cho, D. U., and Cho, Y. H., "Implementation of pre-processing independent of environment and recognition and template matching ", The Journal of the Korean Institute of Communication Sciences, vol. 23, no. 1, pp. 94-100, 1998.

[6] Cornelli, P., Ferragina, P., Granieri. M. N., and Stabile, F., "Optical recognition of motor vehicle license plates", *IEEE Transactions on Vehicular Technology*, vol. 44, no. 4, pp: 790-799, 1995.

[7] Cowell, J., and Hussain, F., "A fast recognition system for isolated Arabic characters", *Proceedings Sixth International Conference on Information and Visualisation, IEEE Computer Society*, London, England, pp. 650-654, 2002.

[8] Cowell, J., and Hussain, F., "Extracting features from Arabic characters", *Proceedings of the IASTED International Conference on COMPUTER GRAPHICS AND IMAGING*, Honolulu, Hawaii, USA, pp. 201-206, 2001.

[9] Cui, Y., Huang, Q., "Automatic license extraction from moving vehicles", *Proceedings of International Conference on Image Processing*, pp. 126-129, 1997.

[10] Devijver, P.A., and Kittler, J., "Pattern Recognition : A Statistical Approach". *Prentice – Hall: London* , 1982.

[11] Hamami, L., and, Berkani, D., "Recognition System for Printed Multi-Font and Multi-Size Arabic Characters", The *Arabian Journal for Science and Engineering*, vol. 27, no. 1B, pp. 57-72, 2002.

[12] Hansen, H., Kristensen, A. W., Kohler, M. P., Mikkelsen, A. W. , Pedersen J. M., and Trangeled, M., "Automatic recognition of license plates", *Institute for Electronic System, Aalborg University*, May 2002.

[13] Hi-Tech Solutions, http://www.htsol.com/ , Israel.

[14] Haykin, S., "Neural Networks: A Comprehensive Foundation", Prentice Hall Inc, 1999.

[15] Hontani, H., and Koga, T., "Character extraction method without prior knowledge on size and information", *Proceedings of the IEEE International Vehicle Electronics Conference (IVEC'01)*, pp. 67-72, 2001.

[16] Hu, M. K., "Visual Pattern Recognition by Moment Invariant", *IRE Transaction on Information Theory*, vol IT – 8, pp. 179-187, 1962.

[17] Huang, K.Y., "Syntactic Pattern Recognition for Seismic Oil Exploration", *Series in Machine Perception and Artificial Intelligence*, vol. 46. pp. 1-3, 2002.

[18] Imai, T., "The parking lot management system with recognizing a license plate number automatically", *Proc. 2nd World Congress ITS*, vol. 2, pp. 873-878, 1995.

[19] Jain, A.K., Mao, J., and, Mohiuddin, K., "Artificial Neural Networks: A tutorial", *IEEE Computer special issue on Neural Computing,* pp.1-49, 1996.

[20] Khotanzad, A., and Hong, Y.H., "Invariant image recognition by zernike moments," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 5, pp. 489-497, 1990.

[21] Khotanzad, A., and Hong, Y.H., "Rotation in-variant image recognition using features selected via a systematic method," *Pattern Recognition*, vol. 23, no. 10, pp. 1089-1101, 1990.

[22] Kim, G. M., "The automatic recognition of the plate of vehicle using the correlation coefficient and Hough transform", *Journal of Control, Automation and System Engineering*, vol. 3, no.5, pp. 511-519, 1997.

[23] Kim, K. K., Kim, K. I., Kim, J.B., and Kim, H. J., "Learning based approach for license plate recognition", *Proceedings of IEEE Processing Society Workshop on Neural Networks for Signal Processing*, vol. 2, pp: 614-623, 2000.

[24] Komada, N. et. al., "The latest major ITS activities in japan – Overview, trends and future scope", *Proc. 2nd World Congress ITS*, vol. 5, pp. 2589-2594, 1995.

[25] Lee, E. R., Kim, P. K., and Kim, H. J., "Automatic recognition of a car license plate using color image processing", *IEEE International Conference on Image Processing 1994*, vol. 2, pp.301-305, 1994.

[26] The MathWorks, Inc, The matlab package, http://www.mathworks.com/.

[27] Morel, J., and Solemini, S., "Variational Methods in Image Segmentation", *Birkhauser, Boston*, 1995.

[28] Naito, T., Tsukada, T., Yamada, K., Kozuka, K., and Yamamoto, S., "License plate recognition method for inclined plates outdoors", *Proceedings International Conference on Information Intelligence and Systems*, pp. 304-312, 1999.

[29] Naito, T. Tsukada, T. Yamada, K. Kozuka, K. and Yamamoto, S., "Robust recognition methods for inclined license plates under various illumination conditions outdoors", *Proceedings IEEE/IEEJ/JSAI International Conference on Intelligent Transport Systems*, pp. 697-702, 1999.

[30] Naito, T., Tsukada, T., Yamada, K., Kozuka, K., and Yamamoto, S., "Robust license-plate recognition method for passing vehicles under outside environment", *IEEE Transactions on Vehicular Technology*, vol: 49 Issue: 6, pp: 2309-2319, 2000.

[31] Nieuwoudt, C., and van Heerden, R., "Automatic number plate segmentation and recognition", *Seventh annual South African workshop on Pattern Recognition*, pp. 88-93, IAPR, 1996.

[32] Nijhuis, J.A.G., ter Brugge, M.H., Helmholt, K.A., Pluim, W., Spaanenburg, L., Venema, R.S., and, Westenberg, M.A., "Car license plate recognition with neural

networks and fuzzy logic", Proceedings, *International Conference on Neural Network*, vol. 5, pp. 2232-2236, 1995.

[33] Optasia Systems Pvt Ltd, http://www.singaporegateway.com/optasia/imps , Singapore.

[34] Park, S. H., Kim, K. I., Jung, K., and Kim, H. J., "Locating car license plates using neural network", *IEE Electronics Letters*, vol.35, no. 17, pp. 1475-1477, 1999.

[35] Parking Products, Inc., http://www.parkingproducts.com/, Vehicle Identification System for Parking Areas (VISPA), USA, 2002.

[36] Percerptics, http://www.perceptics.com/lpr.html , northrop grumman information technology, USA.

[37] Rosenblatt, F., " The Perceptron: A probabilistic model for information storage and organization in the brain", *Psychological Review,* vol.65, pp.386-408.

[38] Salagado, L., Menendez, J. M., Rendon, E., and Garcia, N., "Automatic car plate detection and recognition through intelligent vision engineering", *Proceedings of IEEE 33rd Annual International Carnahan Conference on Security Technology*, pp. 71-76, 1999.

[39] Schalkoff, Robert, J., "Pattern Recognition: Statistical, structural and Neural Approaches", *John Willey & Sons, Inc,* 1992.

[40] Smith A.R, "Tint Fill" , *Computer Graphics*, vol. 13, no. 2, pp. 276-283, 1979.

[41] Trier, I.D., Jain, A.K., and, Taxt, T., "Feature Extraction Methods for Character Recognition – A Survey," *Pattern Recognition*, vol. 29, no. 4, pp. 641-662, 1996.

[42] Widrow, B. and Stearns, S.D. "Adaptive Signal Processing", *Prentice Hall,* New Jersey, 1985.

[43] Yan, Dai., Hongqing, Ma., Jilin, Liu., and Langang, Li, "A high performance license plate recognition system based on the web technique, *Proceedings IEEE Intelligent Transport Systems*, pp. 325-329, 2001.

[44] Yoram, and Hofman, O., Hi-Tech Solutions, http://www.htsol.com/Products/Demo.html., 2001.

[45] Yu, M., and Kim, Y. D., "An approach to Korean license plate recognition based on vertical edge matching", *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 4, pp. 2975-2980, 2000.

[46] Zamir recognition system Ltd., http://www.zamir.com/index1.html , Israel, 1993.

VITA

- Mohammed Jameel Ahmed
- Born on June, 05, 1979.
- Received Bachelor of Engineering (B.E) degree in Computer Science & Engineering from Osmania University, Hyderabad, India in 2000.
- Joined King Fahd University of Petroleum & Minerals in September 2001.
- Email: jameel21@ccse.kfupm.edu.sa / jameelupm@yahoo.com